

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โมบายล์คอมเมิร์ซสำหรับสมาร์ตโฟนด้วยเทคโนโลยีอินเทอร์เน็ต (Front End)
Mobile Commerce for Smart phone with technology .NET (Front End)

นายพรชัย คำน้อย
นายวัฒนา รุ่งฤทธิเดช

เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี.....

62730

21 ส.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษามาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมบายล์คอมเมอร์ซสำหรับสมาร์ทโฟนด้วยเทคโนโลยีค็อดเน็ต (Front End)

โดย

นายพรชัย ตันเยี่ยม

นายวัฒนา รุ่งฤทธิเดช

อาจารย์ที่ปรึกษา

ผศ.ดร. สุรินทร์ กิตติธรรมกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2548

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โมบายล์คอมเมิร์ซสำหรับสมาร์ทโฟนด้วยเทคโนโลยีต่อหน้า(Front End)

Mobile Commerce for Smart phone whit technology .NET (Front End)

ผู้จัดทำ

1. นายพรชัย ตันเอี่ยม รหัสนักศึกษา 46015356

2. นายวัฒนา รุ่งฤทธิเดช รหัสนักศึกษา 46015370



ศ. ๑๑๑๑๑๑
(ผศ.ดร. สุรินทร์ กิตติธรรมกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมบายล์คอมเมอร์ซสำหรับสมาร์ตโฟนด้วยเทคโนโลยีค็อทเน็ต(Front End)

นายพรชัย ต้นเยี่ยม	46015356
นายวัฒนา รุ่งฤทธิเดช	46015370
ผศ.ดร. สุรินทร์ กิตติธรรกุล	อาจารย์ที่ปรึกษา
ปีการศึกษา 2548	

บทคัดย่อ

ระบบจัดการข้อมูลด้วยตำแหน่งสถานที่สำหรับอุปกรณ์โมบาย หรือระบบโลเคชันเบส (Location-Based Mobile Information System LMIS) ได้กลายมาเป็นทางเลือกใหม่ที่ประสบความสำเร็จทางหนึ่งสำหรับผู้ใช้อุปกรณ์โมบายภายในกรุงเทพฯ เพราะระบบสามารถให้บริการข้อมูลที่สอดคล้องกับสถานที่ ที่ผู้ใช้กำลังเดินทางอยู่ในสถานะนั้น

โครงการนี้เป็นการสร้าง โปรแกรมระบบโมบายล์คอมเมอร์ซสำหรับสมาร์ตโฟนโดยอาศัยระบบโลเคชันเบสด้วยเทคโนโลยีค็อทเน็ตคอมแพคเฟรมเวิร์ก ซึ่งเป็นการพัฒนาApplication บน Pocket Pc โดยดึงข้อมูลบริการจากเว็บเซอร์วิสมาใช้ โดยระบบดังกล่าวจะเป็นการพัฒนาเป็นโปรแกรมบริการด้านข้อมูลร้านอาหาร มีบริการค้นหาที่ตั้งของร้าน รายการเมนู โปรโมชันของร้าน เป็นต้น โดยApplicationดังกล่าว จะเรียกข้อมูลจากเว็บเซอร์วิสมาใช้ ข้อมูลตามพิกัดละติจูดและลองจิจูด มาMap กับแผนที่ในประเทศไทย เพื่อให้ผู้ใช้บริการต้องการทราบว่ามีบริเวณที่ตนยืนอยู่ขณะนี้ มีร้านอาหารอะไรอยู่บ้าง ร้านไหนเป็นที่นิยม มีโปรโมชันใดเป็นพิเศษ มาแสดงผลบน Pocket PC เพื่อให้ผู้ใช้บริการเกิดความสะดวกสบายเพิ่มมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOBILE COMMERCE FOR SMARTPHONE ON DOTNET TECHNOLOGY (FRONT END)

Pornchai Tunaium 46015356

Wattana Rungrittidech 46015370

Asst. Prof Dr. Surin Kittitornkun Advisor

Academic Year 2006

ABSTRACT

Location-based mobile information system (LMIS) has become a new and successful alternative for mobile users especially in Bangkok because it can provide information while the users are traveling in highly congested areas. We have implemented an LMIS for restaurants based on web services technology. The system consists of the front-end user interfaces on mobile smartphones and web browsers, the back-end, and the map server.

This project is concerned with the user interfaces to the LMIS for restaurants on smartphones and web browsers. The smartphones shall be running Microsoft Windows Mobile 2003 or later. The location of the user can be acquired from either the built-in GPS (Global Positioning System) or the Bluetooth GPS.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดีด้วยคำปรึกษาจาก ผศ.ดร. สุรินทร์ กิตติธรรมกุล ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ข้าพเจ้ารู้สึกทราบบ้างในความอนุเคราะห์จากท่านอาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณห้องวิจัย Mobile Computer Lab (MCL) ที่ได้สนับสนุนเครื่องมือ ตลอดจนข้อมูล และหนังสือต่างๆ ที่ใช้ในการทำวิจัย

ขอขอบคุณเพื่อน และ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

นายวัฒนา รุ่งฤทธิเดช

นายพรชัย ดันเอียด

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของ โครงการงาน.....	1
1.2 วัตถุประสงค์ของ โครงการงาน.....	2
1.3 ขอบเขตของโครงการงาน.....	2
1.4 วิธีการดำเนินการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.6 ส่วนประกอบของปริญญาานิพนธ์.....	3
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในโครงการงาน.....	4
2.1 ทฤษฎี .NET เทคโนโลยี.....	4
2.1.1 คีอทเน็ตเฟรมเวิร์ก (.NET Framework).....	4
2.1.2 เริ่มต้นรู้จักกับ ASP.NET.....	9
2.1.3 Microsoft Visual Studio.NET.....	10
2.2 ทฤษฎี Web services.....	11
2.3 ทฤษฎี GPS โปรโตคอล NMEA.....	15
2.3.1 การเชื่อมต่อกับ Hardware.....	16
2.3.2 ชนิดของประโยค.....	17
บทที่ 3 การออกแบบและพัฒนา.....	30
3.1 ภาพรวมของระบบ.....	31
3.2 สำหรับการพัฒนาร่วมของระบบ Pocket PC และ WebApplication.....	32
3.3 ฟังก์ชัน Web Service ที่ใช้งานสำหรับ Pocket PC และ Web Application.....	43
3.4 การติดต่อ Web service ภายในตัวโปรแกรม mobileREST.....	49
3.5 การติดต่อ GPS ภายในตัวโปรแกรม mobileREST.....	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การติดต่อMAPแบบSETELLITEของGOOGLEภายในตัวโปรแกรม mobileREST 56

บทที่ 4 การทดลองและผลการทดลอง.....	61
4.1 ขั้นตอนการ Config Device GPS.....	61
4.2 แบบจำลองที่ใช้ในการทำงานของโปรแกรม บน Emulator.....	62
4.3 สมรรถนะของระบบ.....	70
บทที่ 5 บทวิจารณ์และสรุป.....	71
5.1 บทสรุป.....	71
5.2 ปัญหาที่เกิดขึ้น.....	71
5.3 แนวทางการแก้ปัญหา.....	71
5.4 แนวทางการพัฒนาต่อ.....	71
บรรณานุกรม.....	72



สารบัญตาราง

ตารางที่	หน้า
2.1 การเปรียบเทียบระหว่าง Web Services VS Web Application.....	14
2.2 NMEA 2.0.....	20
2.3 NMEA 1.5 อุปกรณ์บางตัวจะไม่รองรับกับ NMEA เวอร์ชัน 1.5.....	21
2.4 NMEA output messages.....	29



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 แสดงโครงสร้างของ .Net Framework	5
2.2 แสดง diagram ของ .Net Framework.....	6
2.3 ตัวอุปกรณ์GPS	28
2.4 แสดงการเชื่อมต่อกับตัวอุปกรณ์.....	28
3.1 แสดงโครงสร้างของระบบ.....	31
3.2 แสดงคลาสไดอะแกรมของระบบ.....	32
3.3a Class Diagramของร้านอาหาร	33
3.3b Class Diagramของร้านอาหาร.....	34
3.4a Sequence Diagram แสดงตำแหน่งสะพานที่ปัจจุบัน.....	35
3.4b Sequence Diagram ค้นหาร้านอาหารบริเวณใกล้เคียง.....	35
3.4c Sequence Diagram ตรวจสอบตำแหน่งและคูแผนที่ร้านอาหาร.....	36
3.4d Sequence Diagram แสดงข้อมูลภายในร้านอาหารภาพถ่ายร้านอาหาร.....	36
3.4e Sequence Diagram เรียกดูเมนูของร้านอาหาร.....	37
3.4f Sequence Diagram เรียกดูโปรโมชันของร้านอาหาร.....	37
3.4g Sequence Diagram คูแผนที่ภายในตึกอาคาร.....	38
3.4i Sequence Diagram บันทึกร้านอาหารที่ชื่นชอบ.....	38
3.4j Sequence Diagram จองที่นั่งของร้านอาหาร.....	39
3.5 กระบวนการการทำงานของบริการจองที่นั่งร้านอาหาร.....	42
3.6 กระบวนการการทำงานของบริการค้นหาร้านอาหาร.....	43
3.7 Solution Explorer ในส่วนของ Service.....	49
3.8 รูปการเพิ่ม Web Reference จาก Web service กลาง.....	49
3.9 Properties ของ Web service กลาง.....	50
3.10 รูปการเพิ่ม Web Reference จาก Web service ร้านอาหาร.....	52
3.11 Properties ของ Web service กลาง.....	52
3.12 Solution Explorer ในส่วนของ mobileGPS.....	53
3.13 หน้าตาของ control fGPS ที่สร้างขึ้น.....	53
3.14 Solution Explorer ในส่วนของ mobileMAP.....	56
3.15 รูปที่ได้จาก url http://kh0.google.com/kh?n=404&v=4&t=t	57
3.16 แสดงลักษณะการแบ่งส่วนของ texture เพื่อ Zoomเข้าไปดูรายละเอียดภายในส่วนย่อย.....	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.17 รูปที่ได้จากส่วนย่อย r จะได้ url http://kh0.google.com/kh?n=404&v=4&t=tr	58
3.18 รูปที่ได้จากส่วนย่อย trsttqssrrqtqrrrs จะได้ url http://kh0.google.com/kh?n=404&v=4&t=trsttqssrrqtqrrrs	59
4.1 ตัวอุปกรณ์ GPS ที่ใช้ทดลอง.....	61
4.2 หน้าต่างกำหนด Serial port สำหรับ Emulator.....	61
4.3 หน้าต่างเพื่อยืนยันการเชื่อมต่อกับ web service.....	62
4.4 แสดงหน้าต่างในขณะที่ไม่ได้เชื่อมต่อกับ web service.....	62
4.5 หน้าต่าง setting เพื่อกำหนด username password และเชื่อมต่อ GPS Receiver.....	63
4.6 หน้าแสดงผลที่การค้นหาอุปกรณ์ GPS Receiver.....	63
4.7 หน้าแรกของตัวโปรแกรม mobileREST.....	64
4.8 หน้าการค้นหา.....	64
4.9 แสดงรายการร้านอาหารที่ได้จากการค้นหาด้วยคำว่า ตลาดกระบี่ mcl.....	64
4.10 แสดงหน้าต่างการเชื่อมต่อ web service ของร้านอาหาร.....	64
4.11 หน้าแรกของร้านอาหาร.....	65
4.12 แสดงรายการเมนูอาหารของร้าน.....	65
4.13 หน้าโปรโมชัน ของร้านอาหาร.....	66
4.14 หน้าต่างสำหรับการจองโต๊ะร้านอาหาร.....	66
4.15 แสดงหน้าต่างการเชื่อมต่อ map service เพื่อดึงภาพแผนที่.....	66
4.16 แผนที่ในบริเวณที่ไม่มีข้อมูลจาก map service จะแสดงข้อมูลที่เฉพาะของ service ร้านอาหาร.....	66
4.17 แสดงการบันทึกข้อมูลลง Favorite.....	67
4.18 แสดงหน้าการเชื่อมต่อ web service เพื่อดึงข้อมูลแผนที่ ภายในตึกอาคาร.....	67
4.19 แผนที่ ที่มี ตึกอาคาร ซึ่งสามารถคลิกเข้าไปดูร้านอาหารหายในตึกได้โดยแยกเป็นชั้น.....	68
4.20 แสดงตึกอาคาร ซึ่งสามารถคลิกเข้าไปดูร้านอาหารหายในตึกได้โดยแยกเป็นชั้น.....	68
4.21 ภาพแผนที่ภายใน ตึก อาคาร.....	68
4.22 ภาพแผนที่ภายใน ตึก อาคารอีกชั้น.....	68
4.23 หน้าแรกของร้านอาหารบนเว็บไซต์.....	69

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.24 หน้าแผนที่ของร้านอาหารบนเว็บไซต์.....	69
4.25 หน้าร้านอาหารบนเว็บไซต์.....	70



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

จากการที่เราได้เห็นว่าในอนาคตนี้ ระบบการสื่อสารในเครือข่ายอินเทอร์เน็ตเต็มไปด้วยข้อมูลต่างๆเต็มไปหมด ซึ่งอยู่ในรูปแบบของภาษา XML โดยมี web services เป็นตัวดึงข้อมูลข่าวสารในรูปแบบดังกล่าวมาใช้งาน ดังนั้นจึงได้เกิดแนวคิดที่จะพัฒนา Application บน Pocket PC ที่จะช่วยอำนวยความสะดวกให้กับผู้ใช้บริการ โดย Application ที่จะพัฒนาขึ้นมาจะเป็นโปรแกรมที่ดึงข้อมูลจาก web services ขึ้นมาอีกทีหนึ่ง โดยเป็น User Interface ไว้เป็นตัวเชื่อมการติดต่อสื่อสารผ่านผู้ใช้บริการบนเครือข่ายการสื่อสารแบบไร้สาย ซึ่งอยู่ในรูปของ Pocket PC เพื่อทำให้เกิดประโยชน์สูงสุดในการให้บริการ และในทางธุรกิจของตนและของผู้ใช้บริการ โดยตอนนี้ ได้มีการนำเทคโนโลยีต่างๆ ที่มีอยู่ในปัจจุบัน นำมาผสมผสานร่วมกัน ทำให้เกิดการตอบสนองความต้องการในรูปแบบใหม่ๆเกิดขึ้น และจะทำให้เกิดตลาดการสื่อสารแบบใหม่ต่อไป

ในปัจจุบันปัจจัยในการเลือกร้านอาหารสักร้านนั้น เกิดขึ้นจากการที่ผู้ใช้บริการเลือกร้านอาหารที่ใกล้บ้านหรือร้านอาหารที่โฆษณาผ่านสื่อหรือจากคนอื่นที่แนะนำมา โดยที่ผู้ใช้ไม่ได้ตัดสินใจที่จะต้องการร้านอาหารเหล่านั้นเอง ถ้าผู้ใช้บริการต้องการเลือกร้านอาหารเองทำให้ผู้ใช้บริการต้องใช้เวลาในการเดินทางเลือกตามร้านอาหารนั้นๆ ทำให้เกิดความไม่สะดวกสบาย ดังนั้นจึงเกิดแนวคิดที่จะพัฒนา Application บน Pocket PC ขึ้น นั่นคือ ระบบค้นหาร้านอาหารออนไลน์ ซึ่งจากแนวความคิดดังกล่าว จะช่วยให้ผู้ใช้บริการเกิดความสะดวกสบายมากขึ้น โดยที่ผู้ใช้บริการอยากได้ข้อมูลของการบริการแบบไหน เพื่อให้ได้ร้านอาหารตามต้องการ มีราคาถูก อร่อย รสนิยม สภาพบรรยากาศ ซึ่งให้ผู้ใช้บริการสามารถตัดสินใจเลือกร้านอาหารได้ถูกต้องโดยไม่สิ้นเปลืองเวลา โดยอาศัยเทคโนโลยีของ Webservices โดยใช้เทคโนโลยี dot net ในการพัฒนาโปรแกรมที่สามารถดูรายละเอียดร้านอาหาร เมนูอาหาร ราคา โปรโมชั่น การจองที่นั่งของร้านอาหารและ review ของร้านนั้นๆ เพื่อช่วยในการตัดสินใจในการเลือกซื้อต่อไป

1.2 วัตถุประสงค์ของโครงการ

ปริญญานิพนธ์ฉบับนี้มุ่งหวังเพื่อศึกษาระบบ โมบายล์คอมพิวเตอร์สำหรับพ็อกเก็ต พีซี (POCKET PC) ด้วยเทคโนโลยีคือทเน็ตคอมแพคเฟรมเวิร์ก (DOTNET COMPACT FRAMEWORK) ซึ่งเป็นการพัฒนาApplication บน Pocket Pc ติดต่อกับ WebServices โดยที่ตัว Pocket PC ต้องใช้ GPS Receiver ตรวจสอบพิกัดละติจูดและ ลองจิจูด โดยใช้โปรโตคอล NMEA แล้วส่งค่าพิกัดดังกล่าวไปให้เว็บเซอร์วิสแล้วให้เว็บเซอร์วิสดึงพิกัด มาMap กับแผนที่ที่ใช้ Thaimapguideและเรียกข้อมูลบริการจากเว็บเซอร์วิส มาแสดงผลบนApplication บน Pocket PC และแสดงผลบนเว็บ Application ผ่าน Browser โดยใช้เทคโนโลยีASP.NET ของคือทเน็ตเฟรมเวิร์ก (DOTNET FRAMEWORK)

1.3 ขอบเขตของโครงการ

ในปริญญานิพนธ์ฉบับนี้ได้นำเสนอวิธีการสร้างระบบ โมบายล์คอมพิวเตอร์สำหรับพ็อกเก็ต พีซี (POCKET PC) โดยอาศัยระบบโลเคชันเบส โดยอาศัยเทคโนโลยีคือทเน็ตคอมแพคเฟรมเวิร์ก (DOTNET COMPACT FRAMEWORK) ในการพัฒนา Application โดยระบบแบ่งออกเป็น 2 ส่วน คือส่วนแรกแสดงผลบน POCKET PC กับส่วนที่สอง จะแสดงผลบนเว็บ Application โดย Applicationที่จะพัฒนานี้เรียกว่า ระบบค้นหาร้านอาหาร โดยระบบจะแสดงตำแหน่งและแผนที่ ปัจจุบัน สามารถตรวจสอบตำแหน่งและดูแผนที่ของร้านอาหารได้ ค้นหาร้านอาหารบริเวณใกล้เคียง แสดงข้อมูลภายในร้านอาหารภาพถ่ายร้านอาหาร เรียกดูเมนู โปรโมชัน ของร้านอาหาร สามารถบันทึกร้านอาหารที่ชื่นชอบได้ และจองที่นั่งของร้านอาหาร โดยใช้คอมพิวเตอร์ในการจำลองระบบโลเคชันเบสด้วยโปรแกรม Visual Studio .NET 2005 ผ่านตัว Pocket PC 2003SE Emulator

1.4 วิธีการดำเนินการ

1. ศึกษารายละเอียดของระบบ โมบายล์คอมพิวเตอร์ของร้านอาหาร
2. ศึกษาเกี่ยวกับการทำงานของโปรโตคอลNMEA (PROTOCOL GPS)
3. ศึกษารายละเอียดการทำงานของตัว ระบบ Bluetooth GPS Receiver
4. ศึกษา DotNet Compact Framework โดย Visual Studio .NET 2005
5. ศึกษา Asp.Net C# โดย Visual Studio .NET 2005
6. ศึกษาเกี่ยวกับการทำงานของระบบเว็บเซอร์วิส

7. วิเคราะห์ และออกแบบระบบ โมบายล์คอมพิวเตอร์ และ เว็บคอมพิวเตอร์

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่สามารถนำออกเผยแพร่โดยไม่ได้รับอนุญาตจากทางมหาวิทยาลัยฯ หากมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่อฝ่ายวิชาการ โทร. 0-2916-1111

8. พัฒนาโปรแกรม Mobile Restaurant
9. พัฒนา Web Restaurant Asp.Net
10. ทดสอบการทำงานของระบบ
11. สรุปผลการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถสร้างระบบ โมบายล์คอมพิวเตอร์สำหรับPOCKET PC โดยอาศัยระบบ โทเคชันเบส
2. ได้รับความรู้ความเข้าใจเกี่ยวกับกระบวนการทำงานของเว็บเซอร์วิส
3. ได้รับความรู้ความเข้าใจเกี่ยวกับ .Net Framework และ .Net Compact Framework
4. ได้รับความรู้ความเข้าใจเกี่ยวกับ Asp.Net ด้วย C#

1.6 ส่วนประกอบของปฏิญานิพนธ์

ปฏิญานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงการ วัตถุประสงค์ของโครงการ ขอบเขตของโครงการ วิธีการดำเนินการ ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของปฏิญานิพนธ์

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงการ ซึ่งประกอบด้วยทฤษฎีอะไรบ้าง ให้บรรยายทฤษฎีทั้งหมดโดยละเอียด

บทที่ 3 กล่าวถึงชิ้นงานของโครงการนี้ ส่วนที่ได้พัฒนาขึ้น การทำงานของระบบหรือชิ้นงานบรรยายโดยละเอียด

บทที่ 4 กล่าวถึงการทดลองและผลการทดลอง ผลที่ได้จากการจำลองระบบ ผลการทดลองหรือผลการดำเนินงานทั้งหมด

บทที่ 5 เป็นบทวิจารณ์และสรุป ซึ่งกล่าวถึงบทสรุปของโครงการ วิจารณ์สิ่งที่ได้รับจากโครงการ และข้อเสนอแนะสำหรับเป็นแนวทางในการพัฒนาต่อ

บทที่ 2

ทฤษฎีพื้นฐานที่ใช้ในโครงการ

บทนำ

โครงการนี้เป็นการสร้างโปรแกรมระบบโมบายล์คอมพิวเตอร์สำหรับ Pocket PC โดยอาศัยระบบโอเอสเอ็มวีเอสด้วยเทคโนโลยีคือทเน็ตคอมแพคเฟรมเวิร์ก ซึ่งเป็นการพัฒนา Application บน Pocket Pc ติดต่อกับ Web Services โดยที่ตัว Pocket PC ต้อง Support GPS แล้วใช้ GPS Receiver ตรวจสอบพิกัดละติจูดและ ลองจิจูด โดยใช้โปรโตคอล NMEA แล้วส่งค่าพิกัดดังกล่าวไปให้เว็บเซิร์ฟเวอร์แล้วให้เว็บเซิร์ฟเวอร์ดึงพิกัด มา Map กับแผนที่จาก Thaimapguide และเรียกข้อมูลบริการจากเว็บเซิร์ฟเวอร์ มาแสดงผลบน Application บน Pocket Pc และแสดงผลบนเว็บ Application ผ่าน Browser โดยใช้เทคโนโลยี ASP.NET

2.1 ทฤษฎี .NET เทคโนโลยี

.NET (อ่านว่า ดอทเน็ต) คือแนวคิดหนึ่งที่ไม่โครซอฟท์ภูมิใจนำเสนอ โดย .NET ตัวนี้ไม่ได้เกี่ยวข้องกับโดเมนเนมของเว็บใดๆทั้งสิ้น แต่ .NET ตัวนี้ หมายถึง การนำเอาอุปกรณ์ทุกอย่างบนโลกมาเชื่อมโยงต่อกันเหมือนตาข่าย (net = ตาข่าย) ซึ่งหากว่าทำสำเร็จแล้วไม่ต้องนึกเลยว่าไมโครซอฟท์จะได้เป็นเจ้าของเทคโนโลยีโดยไม่ต้องสงสัย แต่เรื่องมันไม่ง่ายอย่างที่คิด เพราะอุปกรณ์ต่างๆเหล่านั้นล้วนถูกออกแบบมาต่างหาก การที่มันจะติดต่อสื่อสารกันรู้เรื่องนั้น ย่อมเป็นเรื่องที่เป็นไปได้ยาก ไมโครซอฟท์เล็งเห็นจุดนี้ จึงได้พยายามที่จะคิดค้นสิ่ง ที่เป็นมาตรฐานขึ้น เพื่อให้อุปกรณ์ทุกชนิดทั่วโลกติดต่อสื่อสารกันได้อย่างรู้เรื่อง จึงไม่ใช่เรื่องแปลกถ้าในอนาคตเราเปิดเว็บไซต์เล่นอินเทอร์เน็ตด้วยอุปกรณ์อื่นๆนอกเหนือจากคอมพิวเตอร์

อย่างที่ได้อธิบายไปแล้วในตอนต้นว่าไมโครซอฟท์ต้องการที่จะสร้างอะไรที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่าง เพื่อให้ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบซึ่งหมายถึงมันมีชื่อว่าจะให้มันเป็นระบบมาตรฐาน ระบบนี้คือ .NET Framework

2.1.1 ดอทเน็ตเฟรมเวิร์ก (.NET Framework)

.Net Framework คือ โครงร่างการพัฒนา โปรแกรมคอมพิวเตอร์ ที่ถูกออกแบบมาเพื่อให้
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
อำนวยความสะดวกในการพัฒนาโปรแกรมสมัยใหม่ ที่ใช้งานในระบบเครือข่าย (Internet,
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งหากมีการนำไปใช้

Intranet, Mobile Devices, ฯลฯ) Bill Gates และ Steve Ballmer ได้บรรยายสรุปวิสัยทัศน์ที่เกี่ยวกับ .Net เอาไว้ 3 ข้อหลัก ๆ ได้แก่

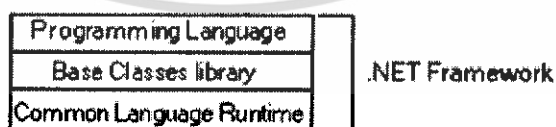
1. การพัฒนาโปรแกรมในรูปแบบของ Web Service จะเป็นหัวใจหลักในการขับเคลื่อนโปรแกรมต่าง ๆ ที่ใช้งานบน Internet. Web Service จะช่วยให้การติดต่อสื่อสารระหว่าง application บน Internet นั้นง่ายขึ้น และเป็นระบบมากยิ่งขึ้น

2. Web Service ขั้นพื้นฐานเช่น การตรวจสอบ user ที่ log in เข้าสู่ระบบ จะถูกพัฒนาให้เป็นมาตรฐาน และสามารถนำไปใช้ได้ทั่วไปบน Internet

3. PC (desktop, notebook) และ Mobile Device ที่ต่อเชื่อมกับ Internet ได้ เช่น PDA และ โทรศัพท์มือถือ จะมีบทบาท และประโยชน์มากขึ้นไปอีก เมื่อสามารถติดต่อใช้งานโปรแกรมต่าง ๆ บน Internet ได้

Microsoft จึงได้พัฒนารูปแบบการพัฒนาระบบคอมพิวเตอร์ขึ้นมา ซึ่งเรียกว่า .Net Framework นั้นเอง (อันที่จริงแล้ว Microsoft ไม่ได้เป็นผู้คิดค้นเรื่องพวกนี้ขึ้นมาแต่เพียงผู้เดียวอย่าเข้าใจผิด สิ่งต่าง ๆ เหล่านี้มีผู้คิดค้นพัฒนาจากหลายบริษัท หลายหน่วยงาน ยกตัวอย่างเช่น Sun Microsystems, IBM, ฯลฯ หากแต่ว่า Microsoft นำแนวคิดเหล่านั้นมาออกแบบให้อยู่ในรูปแบบที่สามารถต่อเชื่อมกันได้ง่ายขึ้น เป็นระบบมากขึ้น) เพราะฉะนั้น ถ้าจะให้นิยามคำว่า .Net คงไม่สามารถชี้เฉพาะได้ว่า .Net คืออะไร เพราะจริง ๆ แล้ว .Net ประกอบไปด้วยส่วนประกอบต่าง ๆ หลายส่วนด้วยกัน ส่วนประกอบเหล่านี้ถูกออกแบบมาเพื่อให้ทำงานได้เข้ากันได้ดียิ่งขึ้น

ระบบนี้ไม่ใช่ระบบปฏิบัติการ (OS) แต่เปรียบเสมือน โปรแกรมหนึ่งที่จะสามารถสร้างสถานะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบ .NET นี้ได้

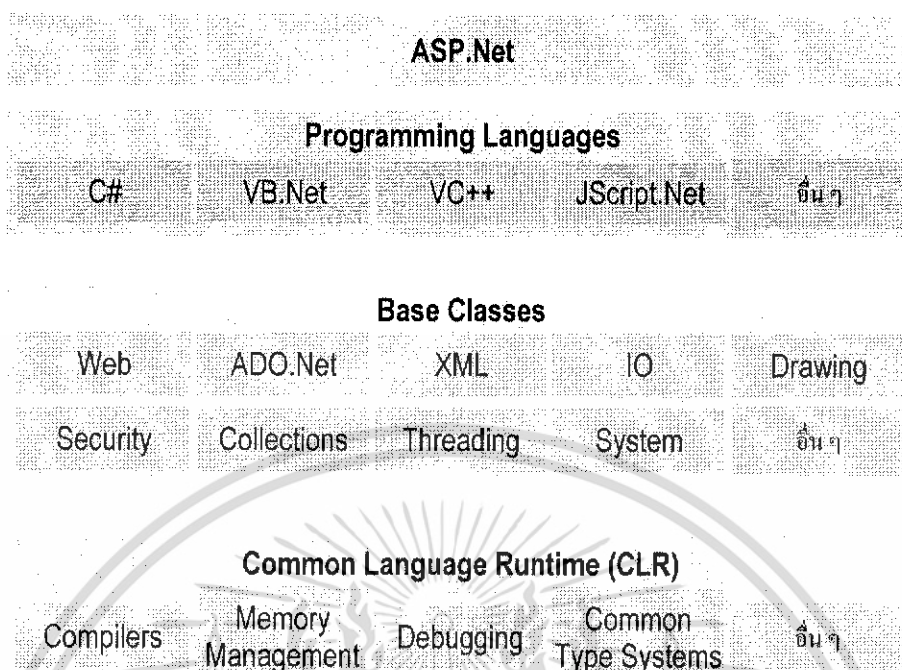


โครงสร้างของ .NET Framework

รูปที่ 2.1 แสดงโครงสร้างของ .Net Framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบหลัก ๆ ของ .Net Framework แบ่งเป็นชั้น ๆ ได้ดัง diagram ต่อไปนี้



รูปที่ 2.2 แสดง diagram ของ .Net Framework

1. Common Language Runtime (CLR) เป็นส่วนพื้นฐานที่ติดต่อกับระบบปฏิบัติการ Windows ทำหน้าที่เป็น run-time environment ให้กับโปรแกรมที่เขียนขึ้นสำหรับใช้บน .Net นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆกัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมดเราเรียกภาษาที่ว่านี้ว่า Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสถานะแวดล้อมการทำงานเช่นใดหลังจากนั้นก็จะมีคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรมต่างๆได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง CLR มีส่วนของ compiler ทั้งที่เป็นแบบปกติ (compile ก่อนที่จะนำโปรแกรมไปใช้) และแบบ Just-In-Time (compile เมื่อจะใช้โปรแกรมนั้น ๆ) มีส่วนของ Memory Management ที่เอาไว้สำหรับจัดสรรหน่วยความจำของเครื่องให้กับโปรแกรม รวมไปถึงการทำ Garbage Collection (การเรียกคืนหน่วยความจำที่ไม่ได้ใช้อีกต่อไป) ส่วนของ Common Type Systems (CTS) ทำให้ภาษาต่าง ๆ ที่เขียนขึ้นบน .Net สามารถทำงานร่วมกันได้ เพราะขนาด และรูปแบบของข้อมูลที่เก็บไว้นั้นเป็นรูปแบบเดียวกัน

2. Base Classes เป็น class library พื้นฐาน ที่โปรแกรมต่าง ๆ ไม่ว่าจะเขียนด้วยภาษาใดบน .Net ก็สามารถใช้ร่วมกันได้ เช่น การติดต่อระบบฐานข้อมูล (ADO.Net), การติดต่อกับ file เอกสารเป็นเอกสารที่ส่งมาไว้สำหรับใช้เพื่อการศึกษาเท่านั้น มิใช่ผู้แต่งเห็นใจใช้ประโยชน์ตามการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

system ของ server (IO), ฯลฯ Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อยๆ ที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆ ส่วนใหญ่จะอยู่ในรูปแบบไฟล์ incode แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ คอมโพเนนต์ต่างๆ นั้นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่ตัวเดียวกันหมด

3. Programming Languages เป็นเซตของ ภาษาคอมพิวเตอร์ ที่ถูกออกแบบมาเพื่อการเขียนโปรแกรมบน .Net Framework โดยที่ทางไมโครซอฟท์ได้เปิดตัวภาษาหลักๆ ที่จะใช้พัฒนาบน .NET นี้ 3 ภาษา

- C# เป็นภาษาใหม่ที่ไมโครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก
- VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0
- JScript.net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชันของ

ไมโครซอฟท์ไมโครซอฟท์นั้นเน้นไปที่ 3-4 ภาษาหลัก ๆ ได้แก่ VB.Net ซึ่งเป็นตัวที่พัฒนาต่อมาจาก VB, C# ซึ่งเป็นภาษาใหม่ที่มี syntax ใกล้เคียงกับ Java และ C++, Visual C++, และ JScript.Net ส่วนภาษาอื่น ๆ นั้น มีบริษัท หรือหน่วยงานอื่น ๆ เป็นผู้พัฒนาขึ้น ซึ่งคาดว่าจะมีเป็นสิบ ๆ ภาษา สำหรับ .Net Framework นั้นไม่ว่าจะเขียนโปรแกรมด้วยภาษาใดก็ตาม Compiler ใน CLR ก็จะ compile โปรแกรมนั้นให้อยู่ในรูปของ Intermediate Language (IL) ซึ่งจะถูกนำไปแปลเป็นภาษาเครื่อง (Native Code) อีกทีเมื่อตอนที่นำไปใช้

4. ASP.Net เป็นภาษา script ที่พัฒนาต่อมาจาก ASP ตัวเก่า เพื่อให้โปรแกรมเมอร์สามารถพัฒนา web application ให้ใช้ .Net ได้สะดวกขึ้น ASP.Net นี้ถึงแม้จะอ้างอิงมาจาก ASP ตัวเก่า แต่ก็มี syntax หลายส่วนที่เปลี่ยนแปลงไป อย่างไรก็ตาม ผู้ที่เคยเขียน ASP มาก่อนไม่น่าจะมีปัญหาในการอ่าน และเขียน ASP มากนัก

หมายเหตุ: สำหรับผู้อ่านที่คุ้นเคยกับ Java ของ Sun มาก่อน จะเห็นได้ว่า .Net Framework กับ Sun's J2EE นั้นคล้ายกันมากทีเดียว สรุปโดยคร่าว ๆ ได้ดังนี้ CLR = JVM (Java Virtual Machine), IL = Java Bytecode, .Net base classes = Java Class Library, .Net Programming Languages = Java Language, ASP.Net = Java Server Page (JSP). ข้อแตกต่างในด้าน Architecture หลัก ๆ ก็คือ IL ของ .Net นั้นต้อง run บน Windows เท่านั้น (ไมโครซอฟท์บอกว่า

run บน OS ไหนก็ได้ แต่อย่างที่ทราบกันดี คงยากที่จะได้เห็นในอนาคตอันใกล้ หรือไม่ก็คงไม่ตีไม่ว่าการณ์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่ากับ run บน Windows) แต่ Java Bytecode นั้นสามารถ run บน OS ไหนก็ได้ที่มี JVM. ส่วนในด้าน performance นั้น ไมโครซอฟท์ได้ทำการเปรียบเทียบโดยพัฒนาโปรแกรม Pet Shop ด้วย .Net โปรแกรม Pet Shop นี้เป็น reference application ที่พัฒนาบน J2EE โดย Sun ไมโครซอฟท์แสดงให้เห็นว่า .Net นั้นทำให้ code สั้นลงหลายเท่าตัว ทำให้โปรแกรมทำงานเร็วขึ้นหลายเท่าเป็นต้น ในส่วนนี้ก็คงต้องรอดูกันต่อไปว่า Sun จะพัฒนา Java อย่างไรเพื่อแก้เกมส์ในจุดนี้

ประโยชน์และข้อดีของ .NET Framework นั้นพอจะสรุปออกมาได้เป็นข้อๆดังนี้

1. เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน : เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้เราไม่ต้องกังวล ว่าภาษาที่ใช้เขียนนั้นมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น

2. ไม่ขึ้นกับระบบปฏิบัติการ (OS) : เนื่องจากระบบปฏิบัติการ ที่แต่ละบุคคลหรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหาเรื่องของเพียงแค่มีระบบ .NET Framework ก็จะทำให้สามารถใช้งาน โปรแกรมต่างๆ ได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่างๆ ได้ทุกระบบปฏิบัติการ

3. ใช้ในการพัฒนาได้ทุกภาษา : ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ๆ เมื่อต้องการสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆ ได้ด้วย

4. มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี : เนื่องจากเป็นระบบที่เป็นมาตรฐาน ทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้น ตลอดโอกาสที่เครื่องจะแอ่งก็ได้เป็นอย่างดี

5. ความปลอดภัยที่มีมากขึ้น : .NET Framework สามารถกำหนดสิทธิ์การใช้งานหรือ permission ของผู้ใช้งานได้มากขึ้นทำให้สามารถกำหนดว่า จะให้โปรแกรมในส่วนใดใช้งานได้หรือไม่ได้ แล้วแต่เฉพาะบุคคล

2.1.2 เริ่มต้นรู้จักกับ ASP.NET

ASP.NET หรืออีกชื่อหนึ่งว่า ASP+ ซึ่งเป็นชื่อที่ไม่โครซอฟท์ใช้เรียกในตอนแรก ถือว่าเป็น ASP เวอร์ชันล่าสุดต่อจาก ASP 3.0 แต่คงไม่สามารถพูดได้เต็มปากว่า ASP.NET พัฒนามาจาก ASP เพราะรูปแบบ และไวยากรณ์ต่างๆ และภาษาที่นำมาใช้งานนั้นต่างจากเดิมแทบทั้งสิ้น แทบจะเรียกได้ว่ายกเครื่องใหม่เลยก็ว่าได้ น่าจะพูดได้ว่า ASP.NET เป็นอีก Generation หนึ่งของ ASP มากกว่า เรามาลองดูกันว่าใน ASP.NET นั้นมีอะไรที่แตกต่างจาก ASP รุ่นก่อน ๆ บ้าง

1) ใช้ภาษาใดๆ ในการเขียนสคริปต์ก็ได้ : จากเดิมที่เราสามารถใช้ได้เฉพาะภาษาที่เป็นสคริปต์ของ VBScript และ JScript แต่ใน ASP.NET เราสามารถใช้ภาษาที่มีรูปแบบของภาษาเต็มๆ ซึ่ง ในเบื้องต้น มี 3 ภาษาคือ C#, VB.NET และ JScript.Net ที่ออกมาเป็นมาตรฐาน แต่ในอนาคตไมโครซอฟท์มีแผนที่จะเพิ่มตัวแปลภาษาให้ครบทุกภาษา

2) มีความยืดหยุ่นในการเขียนโปรแกรมมากขึ้น : โดยที่เราสามารถใช้ภาษาในการเขียน ASP.NET ได้มากกว่า 1 ภาษาภายในไฟล์เดียวกัน ทำให้สามารถเลือกรูปแบบของภาษาที่ง่ายที่สุดต่อการเขียน ในแต่ละส่วนได้

3) ลักษณะการแปลภาษาและนามสกุลไฟล์เปลี่ยนไป : ใน ASP เวอร์ชันก่อนๆ มีลักษณะการแปลภาษาเป็นแบบอินเตอร์พรีเตอร์ (Interpreter) คือการจะทำคำสั่งใดค่อยแปลคำสั่งนั้น แต่ในเวอร์ชัน .NET นี้จะมี ลักษณะเป็นคอมไพเลอร์ (Compiler) คือการแปลคำสั่งรวมทั้งโปรแกรม นอกจากนี้นามสกุลของไฟล์ก็มีการเปลี่ยนแปลงจากเดิมที่ใช้นามสกุลไฟล์เป็น " *.asp " เป็น " *.aspx "

4) รูปแบบและการใช้งานคอมโพเนนต์ที่ง่ายขึ้น : รูปแบบของคอมโพเนนต์จะเน้นไปที่ XML มากที่สุด และที่สำคัญคือการใช้งานคอมโพเนนต์ใน ASP.NET นั้นเราสามารถอัปโหลดไฟล์ไปไว้ในไคลเอนต์ที่ผู้ดูแลเซิร์ฟเวอร์ (Admin) กำหนดหลังจากนั้นคอมโพเนนต์จะติดตั้งตัวเองโดยอัตโนมัติ ลดปัญหาที่เกิดจาก ASP เวอร์ชันก่อนๆ ได้เป็นอย่างดี เนื่องจากใน ASP เวอร์ชันก่อนนั้นการติดตั้งคอมโพเนนต์กระทำได้เพียงผู้ดูแลเซิร์ฟเวอร์เพียงคนเดียวเท่านั้น ทำให้เวลาต้องการใช้คอมโพเนนต์ต่างๆ ที่เซิร์ฟเวอร์ไม่มี จึงเป็นเรื่องที่ลำบาก

5) มีไลบรารีให้เลือกใช้ได้มากขึ้น : ใน ASP เวอร์ชันก่อนๆ นั้นแอปพลิเคชันบางอย่างสร้างได้ไม่สะดวกนัก ต้องอาศัยคอมโพเนนต์ต่างๆ มากมาย แต่ใน ASP.NET นั้นได้เพิ่มไลบรารีเอกส่าในส่วนเหล่านี้ให้กลายเป็นพื้นฐานของกรใช้งาน ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) มีคอนโทรลทำให้การใช้งานในบางสิ่งง่ายขึ้น : เป็นส่วนพิเศษที่เพิ่มเติมมาจาก ASP รุ่นก่อนๆที่ไม่มีส่วนที่เรียกว่า คอนโทรล ซึ่งคอนโทรลนี้จะช่วยให้เราสามารถสร้างเว็บไซต์ได้อย่างง่ายและมีประสิทธิภาพมากขึ้น จึงไม่ต้องกังวลว่าบราวเซอร์รุ่นนั้นรุ่นนี้จะรองรับกับภาษาที่เราเขียนหรือไม่

7) สามารถเรียกขอข้อมูลจากเซิร์ฟเวอร์ได้ : ใน ASP เวอร์ชันก่อนๆเซิร์ฟเวอร์สามารถเรียกขอข้อมูลได้จากเครื่องผู้ใช้เท่านั้นแต่ใน ASP.NET เครื่องเซิร์ฟเวอร์สามารถเรียกขอข้อมูลจากเครื่องเซิร์ฟเวอร์ด้วยกันได้

8) ไม่ต้องต่อ Hardware : เนื่องจากเป็นระบบใน .NET Framework ดังนั้นจึงมีคุณสมบัติของ Common Language Runtime (CLR) ทำให้มีการคอมไพล์โปรแกรมเป็นภาษามาตรฐานที่เรียกว่า IL ก่อน ดังนั้นไม่ว่าคุณจะเล่นเครื่องปาล์มหรือโน้ตบุ๊ก PDA ก็ไม่เกิดปัญหา

9) ง่ายต่อการหาจุดผิดพลาดในการเขียนโปรแกรม : หากเป็น ASP รุ่นก่อนเวลาเกิดความผิดพลาด (error) เครื่องจะบอกแค่ว่าเป็นความผิดพลาดชนิดใดบรรทัดไหน แต่ใน ASP.NET นี้เครื่องจะแสดงรายละเอียดที่มากขึ้น พร้อมแนวทางแก้ไข

10) มีการตรวจสอบเหตุการณ์ต่างๆได้ภายในเว็บเพจ : มีการตรวจสอบเหตุการณ์ต่างๆตั้งแต่โหลดหน้าเว็บเพจไปจนถึงปิดหน้าเว็บเพจลง ทำให้เราสามารถเขียนโปรแกรมกำหนดเหตุการณ์ต่างๆได้ง่ายขึ้น

11) แยกส่วนที่เป็น HTML กับ ASP ออกมาอย่างชัดเจน : ในเวอร์ชันก่อนๆส่วนที่เป็น HTML กับ ASP จะเขียนปนกันไปมา แต่ในเวอร์ชันนี้จะแยกส่วนกันอย่างชัดเจนว่าส่วนไหนเป็น HTML และส่วนไหนเป็น ASP

2.1.3 Microsoft Visual Studio.NET

Microsoft Visual Studio.NET คือเครื่องมือในการพัฒนาซอฟต์แวร์สมบูรณ์แบบที่สนับสนุนการใช้ภาษาโปรแกรมหลายภาษาสำหรับการสร้างและประสาน Web services และแอปพลิเคชันที่ใช้ XML เข้าด้วยกันช่วยเพิ่มศักยภาพให้กับการทำงานของนักพัฒนา และก่อให้เกิดโอกาสใหม่ๆ ทางธุรกิจ โดยได้รับการออกแบบให้รวมเข้ากับมาตรฐาน และ โปรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของอินเทอร์เน็ต เช่น XML และ SOAP อย่างแน่นอน Visual Studio.NETจึงทำให้วงจร ของการพัฒนาแอปพลิเคชันง่ายขึ้นอย่างมาก

เป็นชุดเครื่องมือพัฒนาที่ครอบคลุมสำหรับการสร้างแอปพลิเคชันทุกรูปแบบ ซึ่งมีส่วนติดต่อกับผู้ใช้ (User Interface) ที่หลากหลายเช่น สำหรับ rich client , บราวเซอร์และอุปกรณ์ไร้สาย ซึ่งสิ่งสำคัญต่อนักพัฒนานบนแพลตฟอร์ม .NET ก็คือ นักพัฒนาสามารถจะใช้งานได้โดยเรียนรู้อีกเพียงเล็กน้อย หรืออาจไม่ต้องเรียนรู้เพิ่มเติมเลยใช้ทักษะที่มีอยู่ให้เป็นประโยชน์และเลือกใช้ภาษาโปรแกรมจำนวนมากที่สนับสนุน .NET เพื่อให้เกิดประสิทธิผลสูงสุด คุณสามารถนำแอปพลิเคชันที่ถูกเขียนด้วยภาษาต่างๆมาใช้งานร่วมกัน หรือนำมาใช้ใหม่ ทำให้มั่นใจได้ว่านักพัฒนาสามารถนำชุดคำสั่งเดิมที่มีอยู่มาใช้ได้ และสร้างแอปพลิเคชันได้เร็วขึ้น นอกจากนี้ยังช่วยลดจำนวนชุดคำสั่งที่ต้องเขียนได้ ด้วยการลากและวางองค์ประกอบต่างๆ อีกด้วย ซึ่ง .NET Framework จะเตรียมการสำหรับ XML Web services ให้โดยอัตโนมัติ คุณจึงมุ่งไปที่การเขียนชุดคำสั่งที่เกี่ยวกับการทำงานในธุรกิจได้อย่างเต็มที่ แทนที่จะเสียเวลากับการเขียนชุดคำสั่งด้านโครงสร้างพื้นฐาน ด้วยรูปแบบการเขียนโปรแกรมที่สอดคล้องกันเช่นนี้จึงทำให้การพัฒนาง่ายขึ้น

ทีมพัฒนาสามารถใช้สถาปัตยกรรมแอปพลิเคชันร่วมกัน เริ่มการพัฒนาได้อย่างรวดเร็วและก่อให้เกิดการปฏิบัติงานที่ดีที่สุด Visual Studio .NET ได้รวมกระบวนการพัฒนาทั้งหมดเข้าไว้ด้วยกัน ซึ่งได้แก่ การออกแบบ ทดสอบ นำแอปพลิเคชันไปใช้งาน รวมทั้งการประสานงานระหว่างทีมงานต่างๆ จึงเหมาะสำหรับการพัฒนาในองค์กรขนาดใหญ่

ครอบคลุมวงจรการพัฒนาทั้งหมด Visual Studio .NET เป็นแพลตฟอร์มทรงพลังสำหรับระบบพัฒนาของผู้ผลิตอื่นๆ โดยได้เตรียมการเรียกใช้เครื่องมือในการพัฒนาจำนวนมากภายใต้สภาพแวดล้อมที่เป็นหนึ่งเดียว

2.2 ทฤษฎี Web services

Web Services คือ application หรือ program ที่ทำงานอย่างใดอย่างหนึ่ง ในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจาก application อื่นๆ ในรูปแบบ RPC(Remote Procedure Call) ซึ่งการให้บริการจะมีเอกสารที่อธิบายคุณสมบัติของบริการกำกับไว้ โดยภาษาที่ถูกใช้เป็นตัวในการแลกเปลี่ยนคือ XML ทำให้เราสามารถเรียกใช้ component ใด ๆ ก็ได้ ใน ระบบ หรือ platform ใด ๆ ก็ได้ บน protocol HTTP ซึ่งเป็น protocol สำหรับ World Wide Web หรืออินเทอร์เน็ต อันเป็นช่องทางที่ได้รับการยอมรับทั่วโลกในการติดต่อสื่อสารกันระหว่าง application กับ application ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ Web Services ประกอบไปด้วย มาตรฐานหลัก 4 อย่าง ซึ่งสามารถอธิบายอย่างง่าย ๆ ได้ดังนี้

1. XML (Extensible Markup Language) เป็นภาษามาตรฐานที่ทุกระบบสนับสนุน ทำให้ข้อมูลที่มีโครงสร้างของภาษา XML จะถูกนำไปประมวลผลต่ออย่างอัตโนมัติได้อย่างง่ายดาย ภาษา XML จึงถูกนำมาใช้เป็นภาษามาตรฐานในการแลกเปลี่ยนข้อมูลของ Web Services

2. SOAP (Simple Object Access Protocol) หรือโซฟ เป็นมาตรฐานของเทคโนโลยี Distributed Objects แบบหนึ่ง โดยทำหน้าที่ส่งข้อมูลผ่านอินเทอร์เน็ต ในรูปแบบของ XML ทำให้เรียกใช้งาน โปรแกรมข้ามระบบผ่านทางอินเทอร์เน็ตได้ ซึ่งเป็น protocol ที่ใช้เรียก component ต่าง ๆ ใน platform ใด ๆ ก็ได้ เนื่องจากใช้ XML เป็นรูปแบบของข้อมูลที่ส่ง โดย protocol นี้จะทำงานอยู่บน HTTP ซึ่งเป็น protocol มาตรฐานสำหรับ web อยู่แล้ว ซึ่งนั่นหมายความว่า การเรียกใช้ component โดยอาศัย SOAP ก็คือการเรียกผ่าน web นั้นเอง

อย่างไรก็ดี SOAP ก็ยังมีข้อเสียอยู่บ้าง ได้แก่

1. เนื่องจาก SOAP message นั้นเก็บอยู่ในรูปแบบ XML ทำให้เสียเวลาในการแปลเอกสาร XML กลับมาเป็นรูปแบบที่โปรแกรมเข้าใจ

2. SOAP ทำงานอยู่กับ HTTP protocol ซึ่งมีความเร็วในการรับ และส่งข้อมูล ค่อนข้างต่ำ ด้วยเหตุนี้ทำให้ความเร็วของ SOAP อยู่ในระดับเดียวกับ HTTP ในขณะที่ protocol อื่นเช่น FTP มีความเร็วสูงกว่ามาก

3. WSDL (Web Services Description Language) เป็นภาษามาตรฐานที่ใช้สำหรับอธิบายการใช้งาน โปรแกรมที่เปิดให้บริการ ซึ่งเขียนขึ้นตามแบบมาตรฐาน XML ดังนั้น WSDL จึงเป็นเสมือนคู่มือให้กับระบบ เพื่อเรียนรู้วิธีการเรียกใช้งาน Web Services

WSDL เกิดจากความร่วมมือระหว่าง IBM และ Microsoft WSDL เป็นภาษาที่ใช้อธิบายคุณลักษณะของ web service และวิธีการติดต่อกับ web service นั้น ๆ โดยใช้ไวยากรณ์ของภาษา XML ซึ่ง WSDL อยู่ในความดูแลของ W3C version ล่าสุด คือ WSDL 1.1 ซึ่งหากผู้อ่านสนใจในรายละเอียด สามารถหาอ่านเพิ่มเติมได้จากเว็บไซต์ของ W3C ที่ <http://www.w3.org/TR/wsdl> ส่วนในทางปฏิบัติ หากเราต้องการ สร้าง web service ขึ้นมาเป็นของตนเอง ก็สามารถสร้างเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสาร WSDL ได้โดยอัตโนมัติ เราจึงไม่ต้องไปกังวลในรายละเอียด ในข้อกำหนดใน WSDL มากนัก

4. UDDI (Universal Description, Discovery, and Integration) เป็นมาตรฐานที่จัดตั้งขึ้นโดยบริษัท IBM, Microsoft และบริษัทยักษ์ใหญ่ทางธุรกิจ B2B (Business-to-Business) อื่น ๆ UDDI ถูกสร้างขึ้นมาเป็นระบบมาตรฐานในการอธิบายและค้นหา Web Services โดยเป็นตัวกลางให้ Provider มาลงทะเบียนไว้ โดยใช้ไฟล์ WSDL บอกรายละเอียดของบริษัทและบริการที่มีให้ ทำให้ Requestor สามารถค้นหาและทราบว่าบริษัทมีผลิตภัณฑ์และบริการอะไรบ้าง สามารถติดต่อขอดำเนินธุรกิจการค้ากับบริษัทได้โดยอัตโนมัติผ่านทาง Web Services สำหรับคู่ค้าทางธุรกิจ ซึ่งเปรียบได้กับฐานข้อมูล ขนาดใหญ่ ซึ่งมีข้อมูลของ web service ที่เปิดให้บริการ โดยที่ web site สำหรับค้นหา web service มีอยู่หลายที่อย่างเช่น

<http://uddi.microsoft.com/search.aspx>

<http://www-3.ibm.com/services/uddi/testregistry/find>

<http://uddi.org>

จากมาตรฐานทั้ง 4 อย่างที่กล่าวข้างต้นสามารถสรุปลำดับขั้นตอนการทำงานของ Web Services ได้ดังนี้

1. Provider จัดทำระบบหรือบริการที่เป็น Web Services ขึ้นมา
2. ทำการลงทะเบียน Web Services กับหน่วยงานที่ให้บริการระบบ UDDI (หรือ Registry)
3. นำ WSDL ไฟล์ไปไว้ในระบบ UDDI ที่ได้ลงทะเบียนไว้
4. Requestor ทำการค้นหาระบบหรือบริการที่ต้องการจากระบบ UDDI
5. เมื่อ Requestor ได้พบระบบหรือบริการที่ต้องการจะนำไฟล์ WSDL ไปเรียนรู้วิธีการเรียกใช้ผ่านระบบของตน
6. Requestor ทำการติดต่อและเรียกใช้ระบบหรือบริการจาก Provider ได้โดยตรงผ่าน SOAP ในระบบของตน

ถ้าพิจารณาลักษณะการทำงานและความสามารถของ Web Services แล้ว เราสามารถสรุปได้ว่า Web Services ก็คือ วัฒนาการอีกก้าวหนึ่งของ Web Application นั่นเอง

เปรียบเทียบระหว่าง Web Application และ Web Services

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราเข้าใจความหมายและการทำงานของ Web Application และ Web Services แล้ว จะเห็นว่าเครื่องมือทั้งสองต่างใช้ HTTP Protocol หรือ อินเทอร์เน็ต เป็นช่องทางในการสื่อสาร เหมือนกัน แต่มีวัตถุประสงค์ต่างกัน โดย Web Application ใช้เพื่อการแลกเปลี่ยนไฟล์ HTML ระหว่างเว็บเซิร์ฟเวอร์ แต่ Web Services เป็นการแลกเปลี่ยน “บริการ” (ก้อนโปรแกรม Software Components) ระหว่างระบบสารสนเทศผ่านเว็บเซิร์ฟเวอร์

ในเรื่องของความสามารถโดยส่วนใหญ่จะใช้ Web Application ในการติดต่อกับผู้ใช้ผ่านทาง Internet browser เพื่อนำเสนอข้อมูลและการทำธุรกรรมต่างๆ ส่วน Web Services จะทำหน้าที่ในการติดต่อกับเว็บเซิร์ฟเวอร์เพื่อแลกเปลี่ยนข้อมูลและการทำงานหรือใช้บริการข้ามระบบกันโดยใช้ Web Application หรือ Application Interface ในการติดต่อกับผู้ใช้ นอกจากนี้ Web Services ยังสามารถทำงานกับระบบต่างๆ ได้มากกว่า 1 ระบบ ในขณะที่ Web Application ไม่สามารถทำได้โดยตรง ซึ่งสามารถสรุปการเปรียบเทียบได้ดังตาราง

ตารางที่ 2.1 การเปรียบเทียบระหว่าง Web Services VS Web Application

หัวข้อ	Web Services	Web Applications
การเชื่อมต่อ	program-program	human-program
ภาษาที่ใช้	XML	HTML
รายชื่อการให้บริการ	ค้นหาผ่าน UDDI	ค้นหาผ่าน search engine
ขอบเขตการใช้งาน	Business-to-Business (B2B)	Business-to-Customer (B2C)
โปรโตคอล(Protocol)	SOAP+HTTP	HTTP

หลังจากที่ได้ทำความเข้าใจความหมายและข้อเปรียบเทียบระหว่าง Web Application และ Web Services แล้ว ความแตกต่างระหว่างเครื่องมือทั้งสองจะเห็นได้ชัดในเรื่องของความสามารถ แต่ในการนำไปใช้งานจำเป็นจะต้องประยุกต์ใช้เครื่องมือทั้งสองชนิดร่วมกัน เพื่อให้เกิดประสิทธิภาพมากที่สุด

ในเรื่องของความปลอดภัย เนื่องจาก Web Service ทำงานอยู่บน Internet ซึ่งปัจจุบันมีเทคโนโลยีในการรักษาความปลอดภัยมากมายรองรับอยู่แล้ว และ Web Services สามารถผ่านระบบรักษาความปลอดภัย (Firewall) ได้เนื่องจาก SOAP ถูกส่งโดยผ่านโปรโตคอล HTTP นอกจากนี้ Web Services ยังมีระบบรักษาความปลอดภัยตามมาตรฐานของ PKI (Public Key Infrastructure) เช่น MD5 (Message Digest),

SSL (Secure Socket Layer) และ PGP (Pretty Good Privacy) ทำให้มั่นใจได้ว่าการใช้ Web Services เป็นเครื่องมือธุรกิจมีมาตรฐานในการรักษาความปลอดภัยรองรับ

ส่วนเครื่องมือที่ใช้ในการพัฒนา Web Services ก็มีอยู่มากมาย เนื่องจาก Web Services เป็นมาตรฐานที่พัฒนาได้ง่าย เพราะอยู่ในรูปแบบของข้อมูลตัวอักษรหรือ ภาษา XML ทำให้มีชุดเครื่องมือที่สนับสนุนการพัฒนา Web Services มากมาย ซึ่งสามารถแบ่งออกเป็น 4 กลุ่มใหญ่ได้ ดังนี้

1. ชุดเครื่องมือที่สนับสนุนโดย Microsoft ตาม Platform ของ Microsoft .NET Framework
2. ชุดเครื่องมือที่สนับสนุนโดย Sun Microsystems ตาม Platform ของ Sun ONE (Sun Open Net Environment)
3. ชุดเครื่องมือที่สนับสนุนโดย IBM เช่น Web Services Toolkit
4. ชุดเครื่องมืออื่นๆที่สนับสนุน SOAP, XML ทั้งที่เป็น Commercial Product และ Open Source

2.3 ทฤษฎี GPS โพรโตคอล NMEA

ทฤษฎีการกำหนดตำแหน่งด้วยระบบ GPS ระบบดังกล่าวจะบอกตำแหน่งละติจูด ลองจิจูดของทุกจุดบนพื้นโลก โดยรับสัญญาณจากดาวเทียมบอกตำแหน่งซึ่งทำการโคจรอยู่รอบโลกอย่างน้อย 4 ดวง จากทั้งหมด 24 ดวง ระบบ GPS ใช้ดาวเทียมนาฟสตาร์ (NAVSTAR) โดยการส่งคลื่นวิทยุจากดาวเทียมในอวกาศมายังภาคพื้นดิน และใช้ความต่างของเวลาในการรับส่งสัญญาณระหว่างดาวเทียมกับตัวรับสัญญาณ โดยคำนวณหาตำแหน่ง, ความเร็ว และเวลาให้กับผู้ใช้ตามปรกติระบบ GPS จะมีการใส่รหัสเพื่อให้เกิดความผิดพลาดเล็กน้อย ทั้งนี้ เนื่องจากระบบนี้ถูกพัฒนาขึ้นเพื่อให้กระทรวงกลาโหมของสหรัฐอเมริกา นำระบบนี้มาใช้ในทางการทหาร แต่สัญญาณดาวเทียมที่ถูกส่งออกมานั้นสามารถถูกรับได้โดยผู้ใช้ทั่วไป ดังนั้นจึงมีการใส่รหัสเพื่อให้ผู้ใช้ที่ได้รับอนุญาตเท่านั้นใช้งานได้ ในการใช้งานระบบ GPS ในโหมดมาตรฐานจะมีความเที่ยงตรงโดยเฉลี่ย 100 เมตร ในแนวนอนและนอนตั้ง 156 เมตร ในงานที่ต้องการความเที่ยงตรงมากจึงต้องใช้ DGPS (Difference Global Position System) โดยรูปแบบของ DGPS จะประกอบด้วยเครื่องรับที่เรียกว่า “เครื่องรับอ้างอิง” ซึ่งทราบตำแหน่งที่ถูกต้องอย่างแท้จริงอยู่แล้ว ส่วนเครื่องรับ GPS อื่นๆ จะเป็นของผู้ใช้งาน โดยเครื่องรับอ้างอิงหรือสถานีอ้างอิงนี้จะนำสัญญาณดาวเทียม GPS มาคำนวณพิสัยของดาวเทียมแต่ละดวง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตามถ้าพูดถึงมาตรฐาน NMEA จะไม่จำเป็นจะต้องเป็น RS232 แต่จะนิยมใช้ EIA-422 ซึ่งเป็น Interface ที่สามารถปรับความเร็วได้บนโมเดลบางโมเดล โดยที่ NMEA มาตรฐานจะใช้ความเร็ว 4800 baud , 8 bit of data , no parity , one stop bit อุปกรณ์ทุกตัวที่สนับสนุน NMEA จะต้องสนับสนุนความเร็วตามที่กล่าวมา ข้อควรจำ ที่ความเร็ว baud rate 4800 คุณสามารถส่งข้อมูลได้มากกว่าที่ต้องการในช่วงระยะเวลา 1 วินาที ด้วยเหตุผลนี้ อุปกรณ์บางตัวส่งข้อมูลอืดๆ 2 วิหรืออาจจะส่งข้อมูลในทุกวินาที หรือในอุปกรณ์บางตัวส่งข้อมูล 2 ครั้งใน 1 วินาที โดยทั่วไประยะเวลาในการส่งข้อมูลบางส่วนจะสามารถทำเสร็จได้ภายใน 1 วินาที

2.3.2 ชนิดของประโยค

NMEA ประกอบขึ้นจากประโยค โดยที่คำแรกจะแสดงถึงชนิดของข้อมูลเรียกว่า data type ซึ่งจะเป็นตัวระบุรูปแบบของคำที่เหลือที่อยู่ในประโยค ซึ่งแต่ละ data type จะมีความหมายของประโยคแตกต่างกันซึ่งถูกกำหนดโดยมาตรฐาน NMEA

ประโยค GGA ที่แสดงดังตัวอย่างด้านล่าง จะมีรูปแบบข้อมูลที่คงที่ ซึ่งข้อมูลจะถูกส่งมาซ้ำๆในรูปแบบเดิม แต่ตัวข้อมูล data จะเปลี่ยนไป เมื่ออุปกรณ์หรือโปรแกรมทำการอ่านข้อมูลหรือประโยคที่ได้รับ เป็นที่น่าสนใจว่าโปรแกรมสามารถที่จะเลือกรับเฉพาะข้อมูลรูปแบบที่ต้องการได้ และละทิ้งประโยคอื่นๆที่ไม่สนใจ ในมาตรฐาน NMEA จะไม่มีส่วนของคำสั่งที่ระบุว่า gps จะต้องทำงานอย่างไร อุปกรณ์รับสัญญาณ gps บางตัวจะมีคำสั่งภายใน ที่สามารถสั่งให้ส่งข้อมูลมาเฉพาะประโยคที่ต้องการได้

ประโยคของมาตรฐาน NMEA มีหลายชนิดขึ้นอยู่กับชนิดของตัวอุปกรณ์ สำหรับประโยคที่ใช้กับงานทางทะเล ที่ใช้อยู่ในตัว gps มีดังนี้คือ

ทุกประโยคจะขึ้นต้นด้วย GP

- AAM - Waypoint Arrival Alarm
- ALM - Almanac data
- APA - Auto Pilot A sentence
- APB - Auto Pilot B sentence
- BOD - Bearing Origin to Destination
- BWC - Bearing using Great Circle route
- DTM - Datum being used.
- GGA - Fix information
- GLL - Lat/Lon data
- GSA - Overall Satellite data
- GSV - Detailed Satellite data
- MSK - send control for a beacon receiver
- MSS - Beacon receiver status information.
- RMA - recommended Loran data
- RMB - recommended navigation data for gps
- RMC - recommended minimum data for gps
- RTE - route message
- VTG - Vector track an Speed over the Ground
- WCV - Waypoint closure velocity (Velocity Made Good)
- WPL - Waypoint information

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- XTC - cross track error
- XTE - measured cross track error
- ZTG - Zulu (UTC) time and time to go (to destination)
- ZDA - Date and Time

ในเครื่องรับ gps บางตัวจะมีคำสั่งพิเศษคือ

- HCHDG - Compass output
- PSLIB - Remote Control for a DGPS receiver

เวอร์ชันล่าสุดของ NMEA คือ เวอร์ชัน 2.3 ที่ได้เพิ่มการแสดงผลสถานะโหมคการทำงาน
เข้าไปในประโยคหลายประโยค



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 NMEA 2.0

Name	Garmin	Magellan	Lowrance	Notes:
GPAPB	N	Y	Y	Auto Pilot B
GPBOD	Y	N	N	bearing, origin to destination - earlier G-12's do not transmit this
GPGGA	Y	Y	Y	fix data
GPGLL	Y	Y	Y	Lat/Lon data - earlier G-12's do not transmit this
GPGSA	Y	Y	Y	overall satellite reception data, missing on some Garmin models
GPGSV	Y	Y	Y	detailed satellite data, missing on some Garmin models
GPRMB	Y	Y	Y	minimum recommended data when following a route
GPRMC	Y	Y	Y	minimum recommended data
GPRTE	Y	U	U	route data, only when there is an active route. (this is sometimes bidirectional)
GPWPL	Y	Y	U	waypoint data, only when there is an active route (this is sometimes bidirectional)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 NMEA 1.5 อุปกรณ์บางตัวจะไม่รองรับกับ NMEA เวอร์ชัน 1.5

Name	Garmin	Magellan	Notes:
GPAPA	N	Y	Automatic Pilot A
GPBOD	Y	N	bearing origin to destination - earlier G-12's do not send this
GPBWC	Y	Y	bearing to waypoint using great circle route.
GPGLL	Y	Y	lat/lon - earlier G-12's do not send this
GPRMC	Y	N	minimum recommend data
GPRMB	Y	N	minimum recommended data when following a route
GPVTG	Y	Y	vector track and speed over ground
GPWPL	Y	N	waypoint data (only when active goto)
GPXTE	Y	Y	cross track error

รูปแบบของประโยค NMEA 1083 ที่ใช้ในตัว GPS

RMB

\$GPRMB,A,x.x,a,c--c,d--d,lll.l,e,yyyyy.yy,f,g,g,h,h,i,i,j*kk

RMB = Recommended Minimum Navigation Information

- 1 = Data Status (V=navigation receiver warning)
- 2 = Crosstrack error in nautical miles
- 3 = Direction to steer (L or R) to correct error
- 4 = Origin waypoint ID#
- 5 = Destination waypoint ID#
- 6 = Destination waypoint latitude
- 7 = N or S
- 8 = Destination waypoint longitude
- 9 = E or W
- 10 = Range to destination in nautical miles

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 11 = Bearing to destination, degrees True
- 12 = Destination closing velocity in knots
- 13 = Arrival status; (A=entered or perpendicular passed)
- 14 = Checksum

RMC

\$GPRMC,hhmmss.ss,A,llll.ll,a,yyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh

RMC = Recommended Minimum Specific GPS/TRANSIT Data

- 1 = UTC of position fix
- 2 = Data status (V=navigation receiver warning)
- 3 = Latitude of fix
- 4 = N or S
- 5 = Longitude of fix
- 6 = E or W
- 7 = Speed over ground in knots
- 8 = Track made good in degrees True
- 9 = UT date
- 10 = Magnetic variation degrees (Easterly var. subtracts from true course)
- 11 = E or W
- 12 = Checksum

GGA

\$GPGGA,hhmmss.ss,llll.ll,a,yyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh

GGA = Global Positioning System Fix Data

- 1 = UTC of Position
- 2 = Latitude
- 3 = N or S
- 4 = Longitude

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 5 = E or W
- 6 = GPS quality indicator (0=invalid; 1=GPS fix; 2=Diff. GPS fix)
- 7 = Number of satellites in use [not those in view]
- 8 = Horizontal dilution of position
- 9 = Antenna altitude above/below mean sea level (geoid)
- 10 = Meters (Antenna height unit)
- 11 = Geoidal separation (Diff. between WGS-84 earth ellipsoid and mean sea level. -=geoid is below WGS-84 ellipsoid)
- 12 = Meters (Units of geoidal separation)
- 13 = Age in seconds since last update from diff. reference station
- 14 = Diff. reference station ID#
- 15 = Checksum

VTG

\$GPVTG,t,T,,,s.ss,N,s.ss,K*hh

VTG = Actual track made good and speed over ground

- 1 = Track made good
- 2 = Fixed text 'T' indicates that track made good is relative to true north
- 3 = not used
- 4 = not used
- 5 = Speed over ground in knots
- 6 = Fixed text 'N' indicates that speed over ground in in knots
- 7 = Speed over ground in kilometers/hour
- 8 = Fixed text 'K' indicates that speed over ground is in kilometers/hour
- 9 = Checksum

RMA

\$GPRMA,A,llll.ll,N,llll.ll,W,,,ss.s,ccc,vv.v,W*hh

RMA = Navigation data from present position

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1 = Data status
- 2 = Latitude
- 3 = N/S
- 4 = longitude
- 5 = W/E
- 6 = not used
- 7 = not used
- 8 = Speed over ground in knots
- 9 = Course over ground
- 10 = Variation
- 11 = Direction of variation E/W
- 12 = Checksum

GSA

\$GPGSA,A,3,19,28,14,18,27,22,31,39,,,,,1.7,1.0,1.3*35

GSA = GPS receiver operating mode, SVs used for navigation, and DOP values.

1 = Mode:

M=Manual, forced to operate in 2D or 3D

A=Automatic, 3D/2D

2 = Mode:

1=Fix not available

2=2D

3=3D

3-14 = IDs of SVs used in position fix (null for unused fields)

15 = PDOP

16 = HDOP

17 = VDOP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GSV

\$GPGSV,4,1,13,02,02,213,,03,-3,000,,11,00,121,,14,13,172,05*67

GSV = Number of SVs in view, PRN numbers, elevation, azimuth & SNR values.

- 1 = Total number of messages of this type in this cycle
- 2 = Message number
- 3 = Total number of SVs in view
- 4 = SV PRN number
- 5 = Elevation in degrees, 90 maximum
- 6 = Azimuth, degrees from true north, 000 to 359
- 7 = SNR, 00-99 dB (null when not tracking)
- 8-11 = Information about second SV, same as field 4-7
- 12-15 = Information about third SV, same as field 4-7
- 16-19 = Information about fourth SV, same as field 4-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างข้อมูลที่ได้รับ

\$GPRMC,183729,A,3907.356,N,12102.482,W,000.0,360.0,080301,015.5,E*6F

\$GPRMB,A,,,,,,,,,V*71

\$GPGGA,183730,3907.356,N,12102.482,W,1,05,1.6,646.4,M,-24.1,M,*,*75

\$GPGSA,A,3,02,,,07,,09,24,26,,,,,1.6,1.6,1.0*3D

\$GPGSV,2,1,08,02,43,088,38,04,42,145,00,05,11,291,00,07,60,043,35*71

\$GPGSV,2,2,08,08,02,145,00,09,46,303,47,24,16,178,32,26,18,231,43*77

\$PGRME,22.0,M,52.9,M,51.0,M*14

\$GPGLL,3907.360,N,12102.481,W,183730,A*33

\$PGRMZ,2062,f,3*2D

\$PGRMM,WGS 84*06

\$GPBOD,T,,M,,*47

\$GPRTE,1,1,c,0*07

\$GPRMC,183731,A,3907.482,N,12102.436,W,000.0,360.0,080301,015.5,E*67

\$GPRMB,A,,,,,,,,,V*71

ตัวอย่างการถอดรหัสข้อมูล

GGA - essential fix data which provide 3D location and accuracy data.

\$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,*,*47

Where:

GGA Global Positioning System Fix Data

123519 Fix taken at 12:35:19 UTC

4807.038,N Latitude 48 deg 07.038' N

01131.000,E Longitude 11 deg 31.000' E

1 Fix quality: 0 = invalid

1 = GPS fix (SPS)

2 = DGPS fix

3 = PPS fix

4 = Real Time Kinematic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	5 = Float RTK
	6 = estimated (dead reckoning) (2.3 feature)
	7 = Manual input mode
	8 = Simulation mode
08	Number of satellites being tracked
0.9	Horizontal dilution of position
545.4,M	Altitude, Meters, above mean sea level
46.9,M	Height of geoid (mean sea level) above WGS84 ellipsoid
(empty field)	time in seconds since last DGPS update
(empty field)	DGPS station ID number
*47	the checksum data, always begins with *

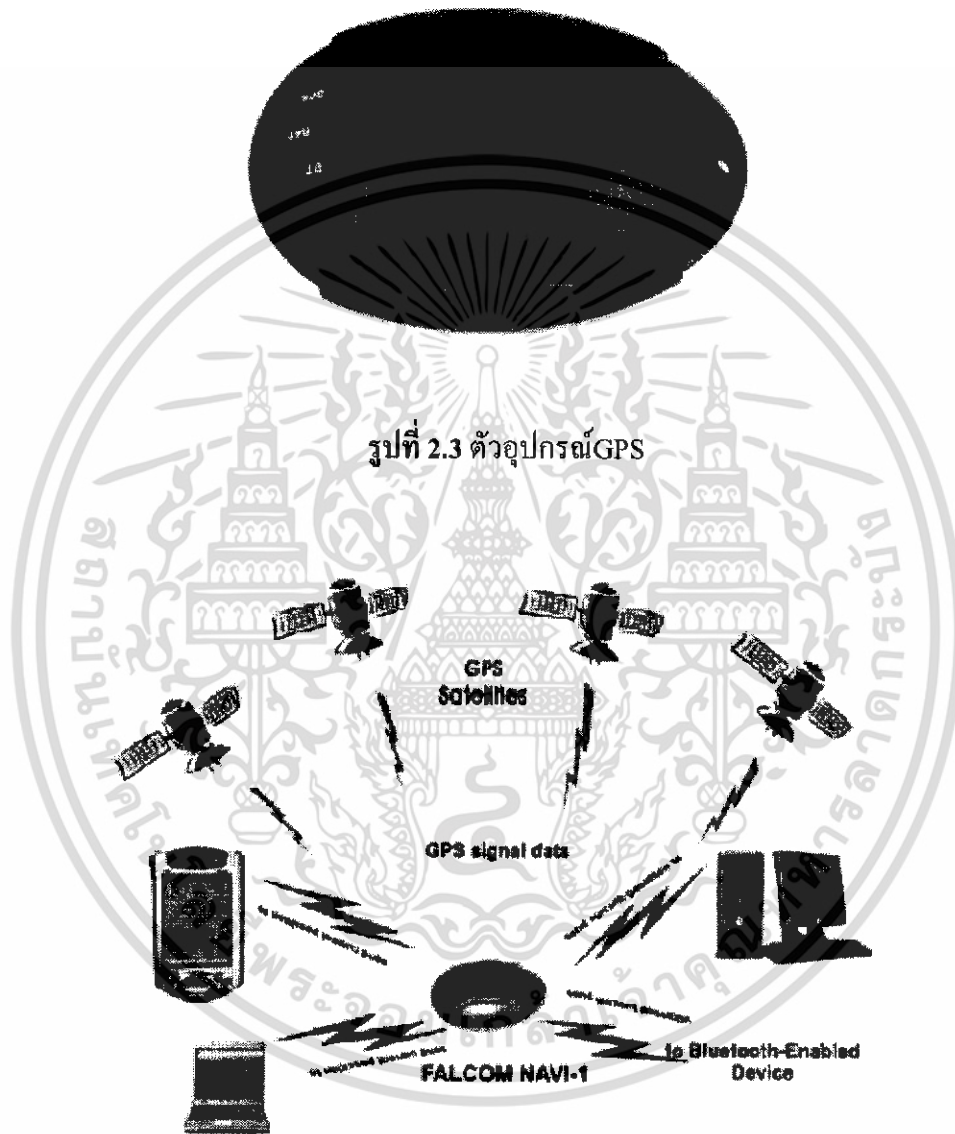


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่ใช้ในการทดลอง

FALCOM NAVI-1

(Bluetooth GPS Receiver)



รูปที่ 2.4 แสดงการเชื่อมต่อกับตัวอุปกรณ์

รูปแบบการเชื่อมต่อกับอุปกรณ์อื่นๆ

ติดต่อผ่านสัญญาณ Bluetooth โดยใช้การจำลองการทำงานแบบ Serial Profile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Interface

อุปกรณ์รับ GPS NAVI-1 จะแสดงข้อมูลในมาตรฐาน NMEA-0183 ที่กำหนดโดย National Marine Electronics Association (NMEA)

NMEA output messages

ตารางที่ 2.4 NMEA output messages

Option	Description
GGA	Time, position and fix type data.
GLL	Latitude, longitude, UTC time of position fix and status.
GSA	GPS receiver operating mode, satellites used in the position solution and DOP values.
GSV	The number of GPS satellites in view satellite ID numbers, elevation, azimuth and SNR values.
RMC	Time, date, position, course and speed data.
VTG	Course and speed information relative to the ground.

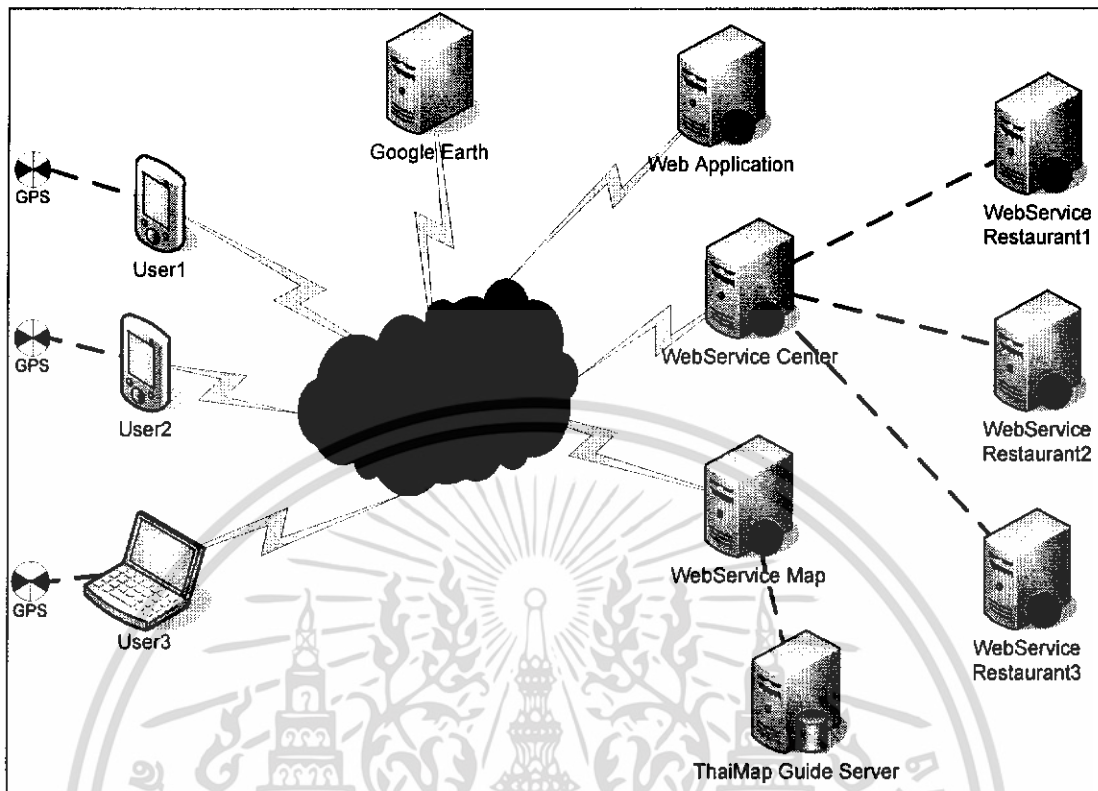
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและพัฒนา

การทำงานของระบบโดยรวมนั้น เริ่มต้นจากผู้ใช้บริการจะใช้บริการผ่าน Pocket PC หรือ PC นั้น ผู้ใช้บริการจะต้องทำการรีจิสเตอร์เพื่อสมัครสมาชิกเพื่อใช้บริการร้านอาหารก่อน โดยสามารถรีจิสเตอร์ได้ผ่านทาง WebApplication โดยจะเรียกใช้ฟังก์ชันจาก Webservices เมื่อได้ทำการสมัครเรียบร้อยแล้วผู้ให้บริการก็สามารถใช้บริการได้โดยผ่านทาง UserInterface ซึ่งออกเป็น 2 ส่วน คือในส่วนแรกเป็นส่วนของโปรแกรม Application แสดงผลบน Pocket PC กับ ส่วนที่ 2 จะเป็นส่วนของ Web Application ซึ่งจะแสดงผลผ่านทาง Web Browser สำหรับ Notebook หรือ PC ทั่วไป สำหรับบริการดังกล่าว ก็จะแยกตามประเภทของผู้ใช้งานว่ามีสิทธิ์เข้าถึงข้อมูลหรือแก้ไขข้อมูลได้มากน้อยเพียงใด ตัวอย่างการให้บริการเช่นผู้ใช้ต้องการทราบว่า ในบริเวณที่ตนเองอยู่มีร้านอาหารใดอยู่บ้าง ผู้ใช้ก็จะเรียกใช้บริการ โดยการส่งข้อมูลที่เป็น ละติจูด, ลองจิจูด และ รัศมีที่ต้องการค้นหาร้านอาหาร ไปยัง Webservices เมื่อ Webservices ได้รับข้อมูลก็จะทำการประมวลผล โดยนำข้อมูลทั้งหมดที่ได้รับเพื่อคำนวณหาร้านอาหารในบริเวณนั้นๆ ทั้งนี้ก็จะทำการติดต่อไปยัง Map server เพื่อทำการขอข้อมูลแผนที่และนำไปมาร์คจุดร้านอาหารซึ่งข้อมูลร้านอาหารทั้งหมดสามารถค้นหาได้จาก Webservices โดยข้อมูลทั้งหมดก็จะถูกส่งกลับไปยังผู้เรียกใช้บริการต่อไป

3.1 ภาพรวมของระบบ



รูปที่ 3.1 แสดงโครงสร้างของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

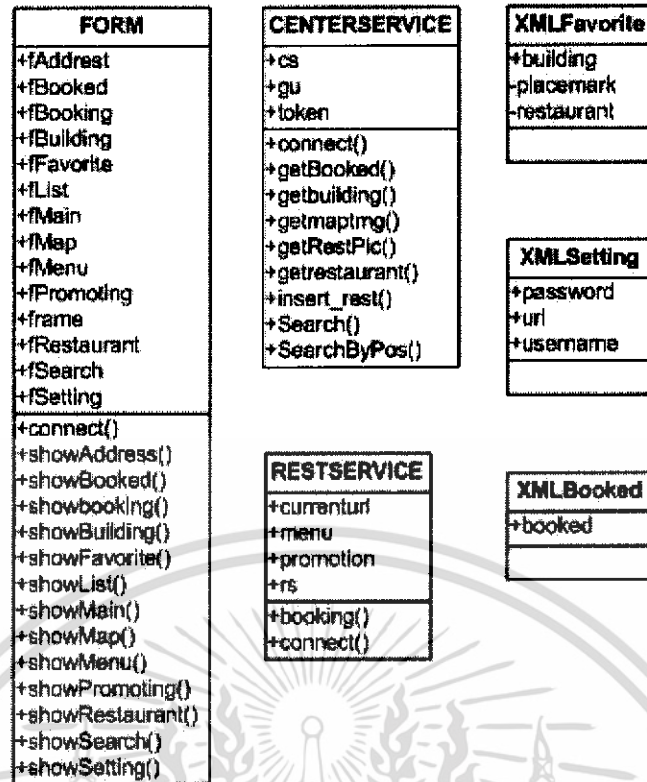
3.2. สำหรับการพัฒนาระบบ Pocket PC

Use Case Diagram



รูปที่ 3.2 แสดงคลาสไดอะแกรมของระบบ

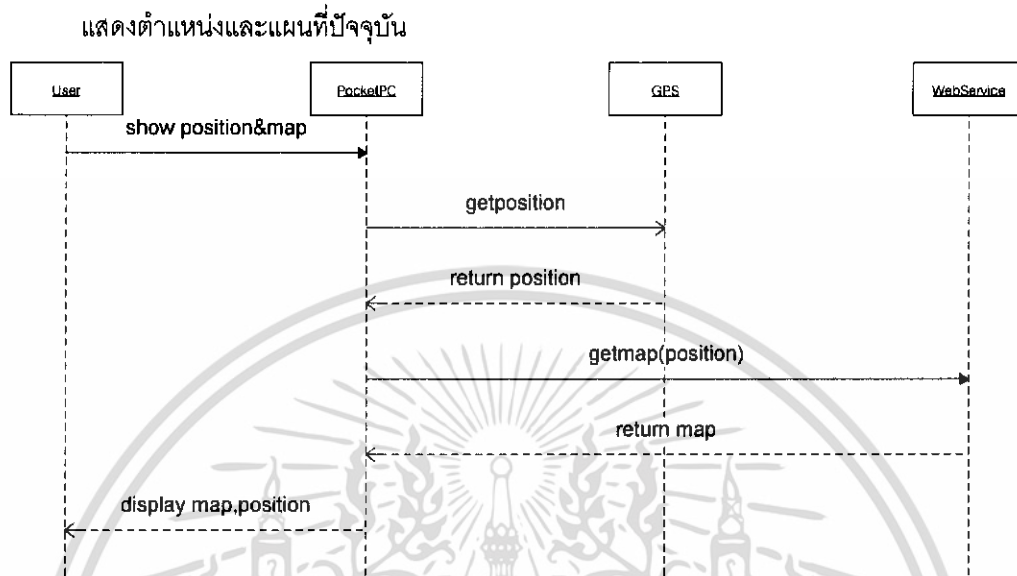
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



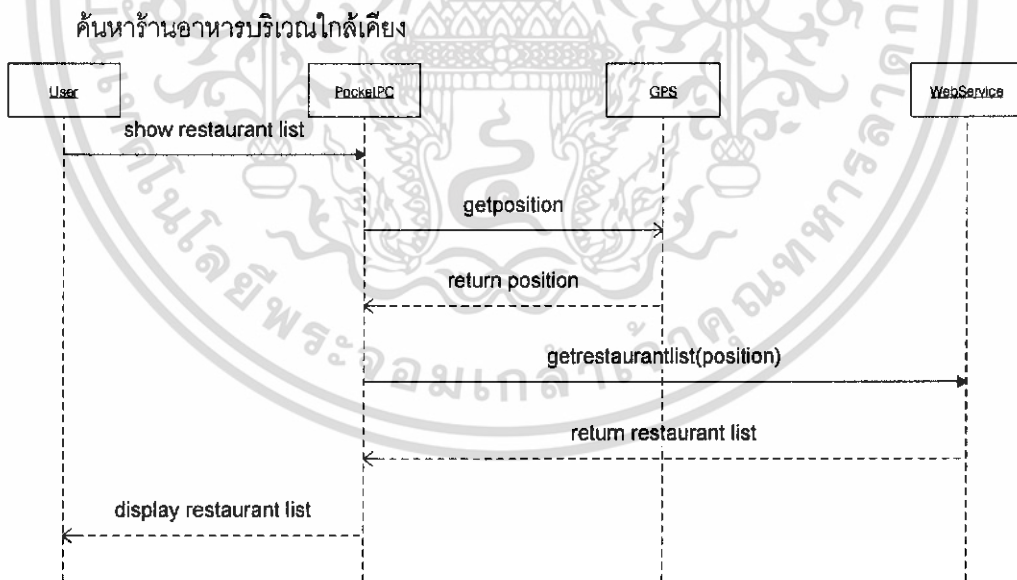
รูปที่ 3.3b Class Diagramของร้านอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sequence Diagram



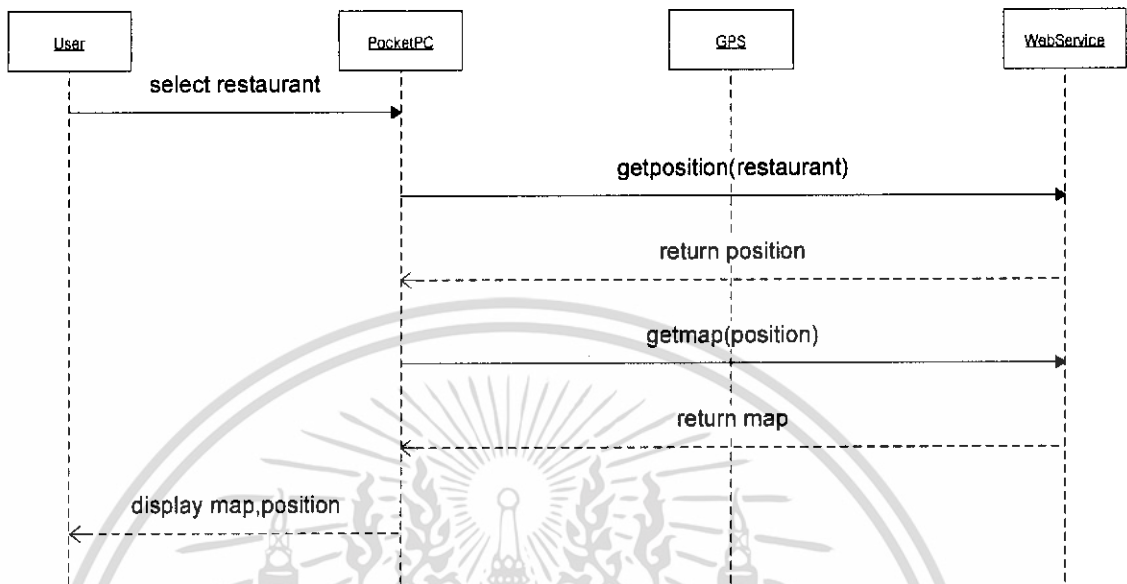
รูปที่ 3.4a Sequence Diagram แสดงตำแหน่งและแผนที่ปัจจุบัน



รูปที่ 3.4b Sequence Diagram ค้นหาร้านอาหารบริเวณใกล้เคียง

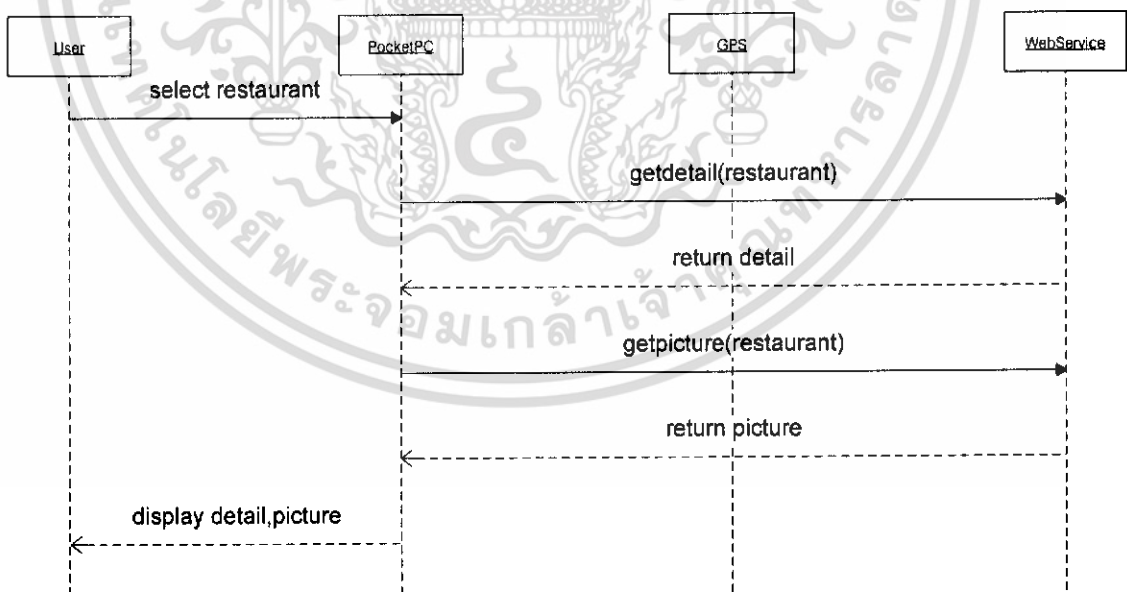
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบตำแหน่งและแผนที่ของร้านอาหาร



รูปที่ 3.4c Sequence Diagram ตรวจสอบตำแหน่งและแผนที่ร้านอาหาร

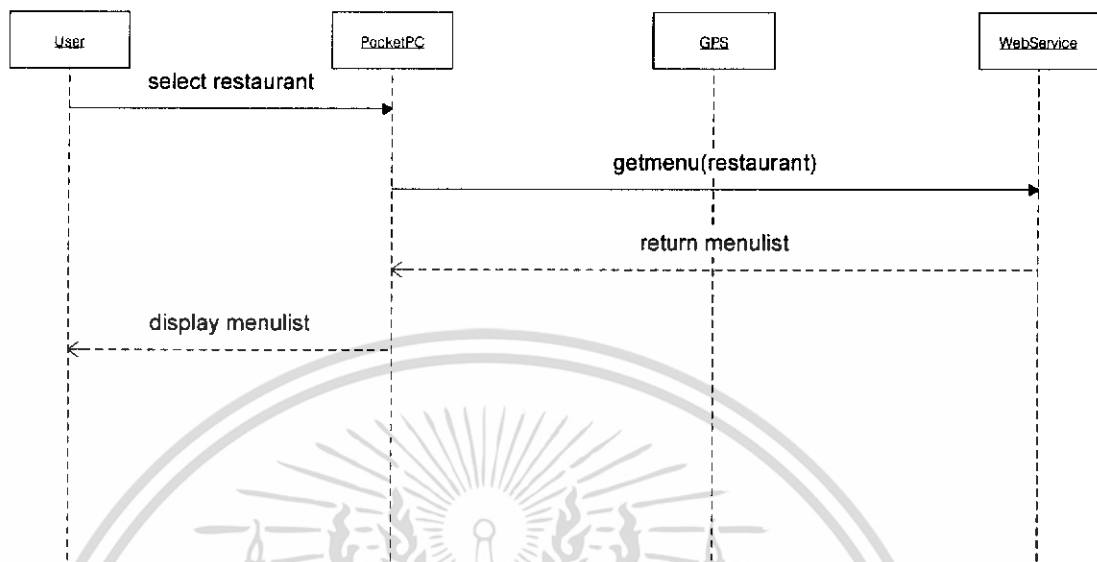
แสดงข้อมูลภายในร้านอาหาร ภาพถ่ายร้านอาหาร



รูปที่ 3.4d Sequence Diagram แสดงข้อมูลภายในร้านอาหารภาพถ่ายร้านอาหาร

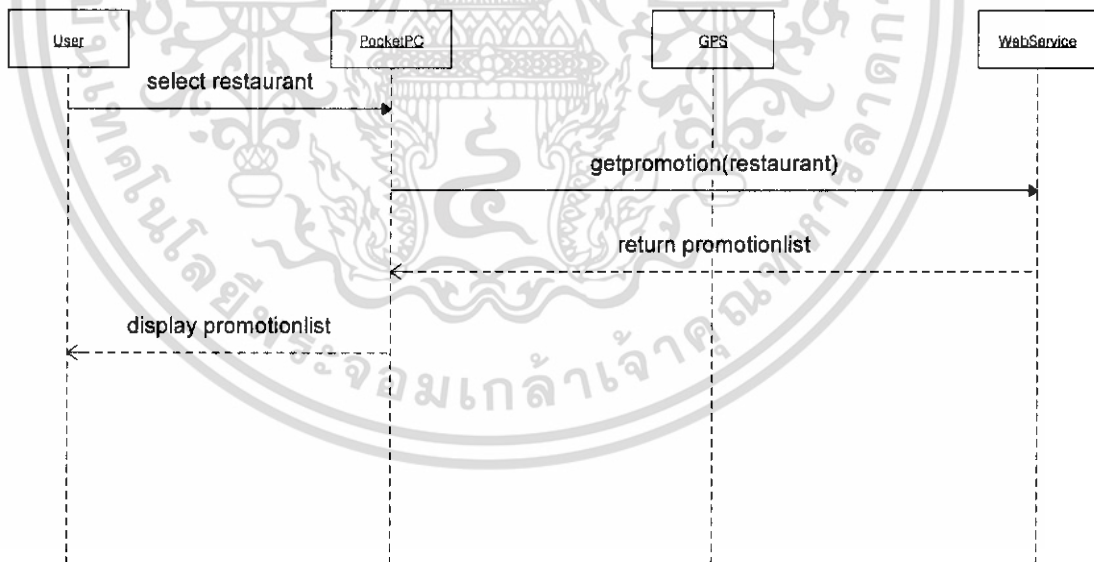
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกดูเมนูของร้านอาหาร



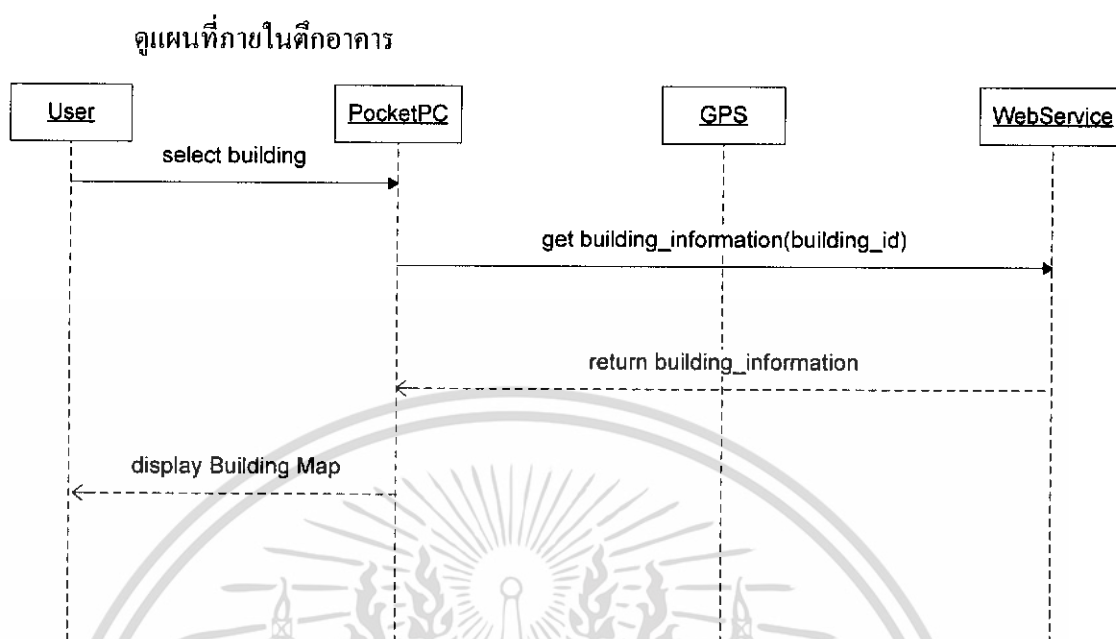
รูปที่ 3.4e Sequence Diagram เรียกดูเมนูของร้านอาหาร

เรียกดูโปรโมชั่นของร้านอาหาร

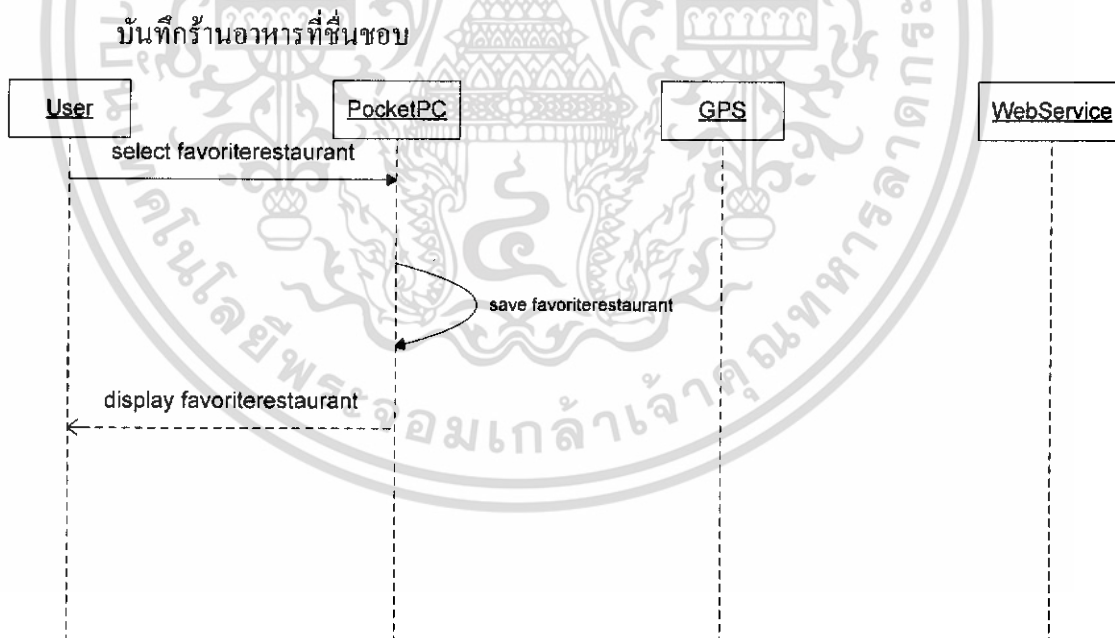


รูปที่ 3.4f Sequence Diagram เรียกดูโปรโมชั่นของร้านอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

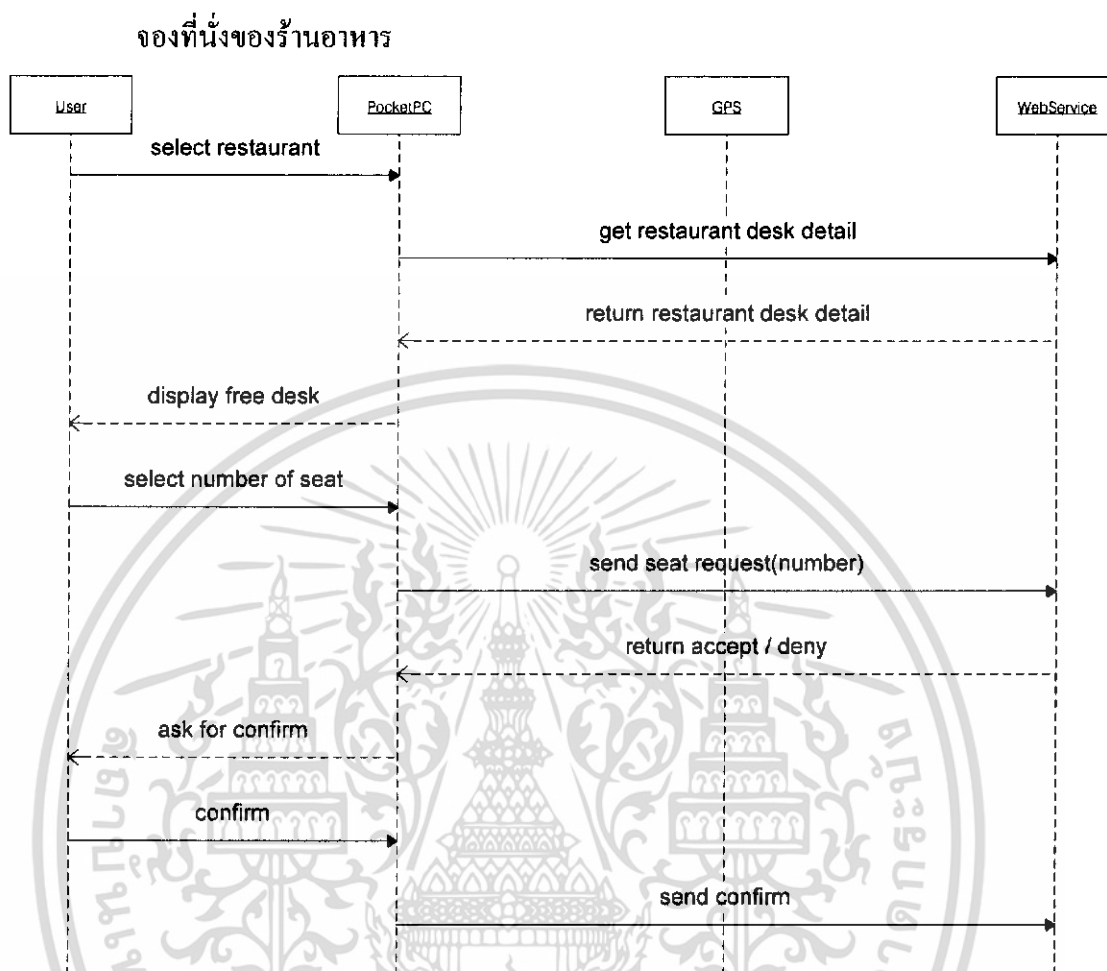


รูปที่ 3.4g Sequence Diagram ดูแผนที่ภายในตึกอาคาร



รูปที่ 3.4i Sequence Diagram บันทึกร้านอาหารที่ชื่นชอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4j Sequence Diagram จองที่นั่งของร้านอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประเภทของผู้ใช้งาน

เราจะแบ่งประเภทของผู้ใช้งานได้ 2 ประเภท โดยสิทธิ์การเข้าถึงข้อมูลจะไม่เท่ากัน โดยการใช้บริการเว็บเซอร์วิสได้ผู้ใช้งานจำเป็นที่จะต้องทำการสมัครสมาชิกโดยหากเป็นนักท่องเที่ยวนักหรือบุคคลทั่วไปก็จะสมัครในประเภทผู้ใช้งานทั่วไป สำหรับอีกประเภทเป็นผู้ใช้งานที่เป็นเจ้าของร้านอาหารที่ต้องการให้ระบบเก็บข้อมูลร้านอาหารของตัวเอง

1. ผู้ใช้งานทั่วไป สำหรับสิทธิ์การใช้บริการได้แก่
 - a. สามารถค้นหาร้านอาหารได้
 - b. สามารถหารายการเมนูอาหารของร้านอาหารนั้นๆได้
 - c. สามารถหารายการโปร โมชั่นของร้านอาหารนั้นๆได้
 - d. สามารถจองที่นั่งร้านอาหารได้
 - e. สามารถเพิ่มร้านอาหารใหม่เข้าเข้าไปได้(จะเป็นประเภทร้านอาหารที่ไม่ยืนยันว่ามีอยู่จริง)
2. ผู้ใช้งานที่เป็นเจ้าของร้าน
 - a. มีสิทธิ์การใช้งานเหมือนผู้ใช้งานทั่วไปทั้งหมด
 - b. สามารถแก้ไขร้านของตนเองได้
 - c. สามารถแก้ไขข้อมูลการจองที่นั่งร้านอาหารของลูกค้าในร้านของตนเองได้

ประเภทของร้านอาหารในระบบ

ประเภทของร้านอาหาร โดยการแบ่งตามประเภทของอาหารที่ขาย(ตัวอย่างประเภทร้านอาหาร)

1. ร้านอาหารไทย
2. ร้านอาหารอิสลาม
3. ร้านอาหารตามสั่ง
4. ร้านอาหารจีน
5. ร้านอาหารยุโรป
6. ร้านก๋วยเตี๋ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับนอร์ใช้งานเพื่อการศึกษานานเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า (ซึ่งข้อมูลประเภทร้านอาหารนี้สามารถได้เพิ่มเติมได้อีกตามการประยุกต์ใช้งาน)

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

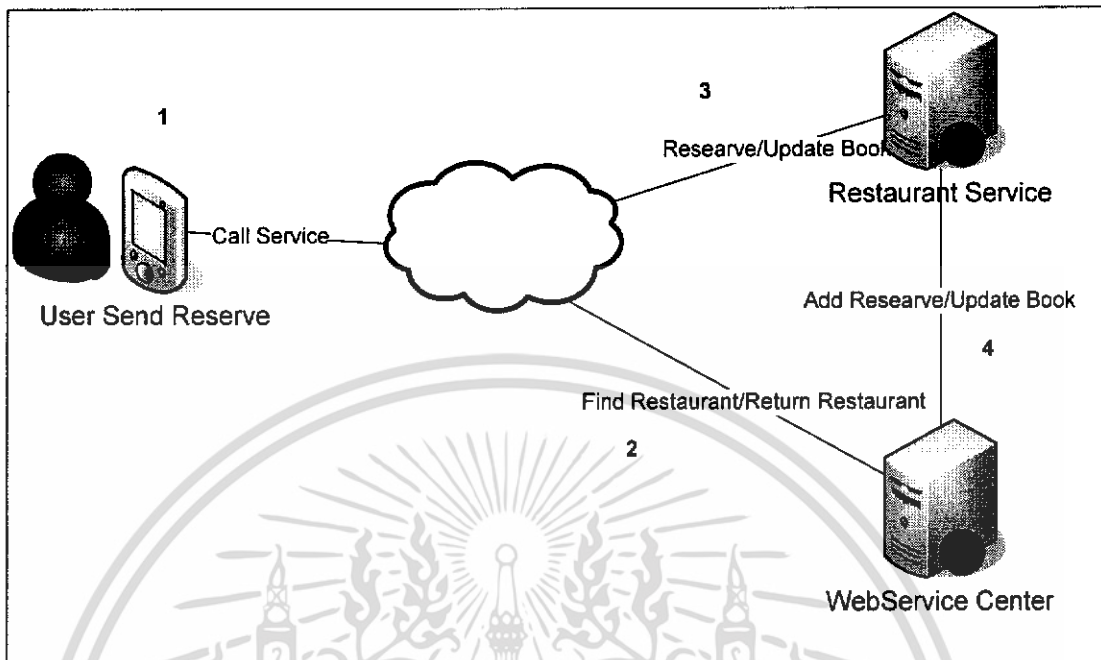
ประเภทของร้านอาหาร โดยการแบ่งตามความน่าเชื่อถือ

1. ร้านอาหารที่ยืนยันว่ามีอยู่จริง โดยร้านอาหารประเภทนี้จะถูกบันทึกชื่อเมื่อเจ้าของร้านมาสมัครใช้บริการ
2. ร้านอาหารที่ไม่ยืนยันว่ามีอยู่จริง ร้านอาหารประเภทนี้จะถูกบันทึกชื่อเมื่อผู้ใช้บริการทั่วไปส่งข้อมูลมาให้กับระบบเหตุที่ระบบไม่ยืนยันว่าร้านอาหารมีอยู่จริงเพราะว่าผู้ใช้อาจทำการส่งข้อมูลมาผิดหรือเกิดจากการกลั่นแกล้ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างกระบวนการทำงาน
การจองที่นั่งร้านอาหาร

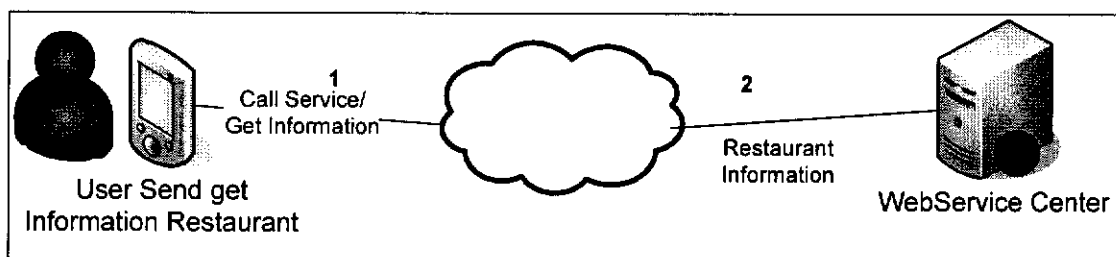


รูปที่ 3.5 กระบวนการการทำงานของบริการจองที่นั่งร้านอาหาร

1. ผู้ใช้บริการค้นหาร้านอาหารที่มีระบบจองที่นั่งผ่าน บริการค้นหาร้านอาหารของ Restaurant Service
2. Restaurant Service ทำการประมวลผลและส่งข้อมูลของร้านอาหารกลับให้แก่ผู้ใช้งาน
3. ผู้ใช้บริการทำการเลือกร้านอาหารที่จะทำการจองหรือทำการแก้ไขข้อมูลที่มีการจองไปแล้วโดยใช้บริการโดยตรงไปยัง Restaurant Service
4. RestaurantServiceทำการบันทึกหรือแก้ไขข้อมูลปัจจุบันของผู้ใช้บริการไปยัง WebserviceCenter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การค้นหาร้านอาหาร



รูปที่ 3.6 กระบวนการการทำงานของบริการค้นหาร้านอาหาร

1. ผู้ใช้บริการเรียกใช้บริการค้นหาร้านอาหาร
2. WebService Center ค้นหาข้อมูลร้านอาหารในบริเวณที่ผู้ใช้ส่งเข้ามาแล้วส่งข้อมูลร้านอาหารทั้งหมดกลับไปให้แก่ผู้ใช้

3.3 ฟังก์ชัน Web Service ที่ใช้งานสำหรับ Pocket PC และ Web Application

ฟังก์ชันสำหรับการเชื่อมต่อ

string checkPWD(string user,string pass)

ใช้สำหรับการยืนยันตัวตนกับ Webservice กลางของระบบ โดยส่ง

Parameter

Username และ Password เป็นแบบ string

Return

string ของ token โดยถ้าการยืนยันสำเร็จ token ที่ได้จะถูกส่งขึ้นมาเป็นกลุ่มของตัวเลข ตัวอักษรและสัญลักษณ์โดยจะไม่มีค่าซ้ำกันในเชื่อมต่อแต่ละครั้ง ค่า token จะถูกส่งเป็น parameter ไปกับฟังก์ชันที่จำเป็นจะต้องมีการยืนยันตัวตน

GeneralUser GetProfile(string user,string pass)

เรียกข้อมูลส่วนตัวของสมาชิก โดยส่ง

Parameter

Username และ Password เป็นแบบ string

Return

GeneralUser ประวัติส่วนตัวของสมาชิก มีข้อมูลดังนี้

- Name : String ชื่อจริงของสมาชิก
- Username : String ชื่อ username ของ สมาชิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Address : String ที่อยู่
- Identify : String รหัสประชาชน

ฟังก์ชันสำหรับค้นหาข้อมูลโดยระบุ keyword

Restaurant[] find_restaurant(string key,string token)

ค้นหารายการร้านอาหาร โดยใช้ keyword

Paramter

key : string คำที่ใช้ค้นหาร้านอาหาร โดยสามารถใช้ key ได้หลายคำโดยใช้เว้นวรรคเป็นตัวขึ้น โดยจะใช้ operator AND ในการค้นหา

token : string token ที่ได้จากการยืนยันตัวตน

Return

Restaurant[] Array รายการของร้านอาหารที่ตรงกับ keyword ใน Restaurant จะประกอบด้วยข้อมูลดังต่อไปนี้

- ID_Restaurant : string รหัสของร้านอาหาร
- Name : string ชื่อร้านอาหาร
- Type : string ชนิดร้านอาหาร
- Telephone : string เบอร์โทร
- Address : ที่อยู่ร้านอาหาร มีข้อมูลย่อยดังนี้
 - Road : string ถนน
 - Amphur : string อำเภอ
 - Zone : string บริเวณที่สำคัญ
 - District : string ตำบล
 - Province : string จังหวัด
 - Zipcode : string รหัสไปรษณีย์
- Position : ตำแหน่งของร้านอาหาร
 - Latitude : string ตำแหน่ง ละติจูด
 - Longitude : string ตำแหน่ง ลองจิจูด
- Valid : string สถานะร้านอาหาร มี
 - V คือร้านอาหารที่มี Webservice
 - N คือร้านอาหารที่ไม่มี Webservice

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและเผยแพร่ไปยังผู้อื่นโดยไม่ได้รับอนุญาตจากผู้นิเทศ

Reserve : string คือ Url ของ Webservice ร้านอาหาร

Building2[] find_building(string key,string token)

ค้นหารายการตึกอาคาร โดยใช้ keyword

Paramter

key : string คำที่ใช้ค้นหาตึกอาคาร โดยสามารถใช้ key ได้หลายคำโดยใช้เว้นวรรคเป็นตัวขึ้น โดยจะใช้ operator AND ในการค้นหา

token : string token ที่ได้จากการยืนยันตัวตน

Return

Building2[] Array รายการข้อมูลตึกอาคารที่ตรงกับ keyword ใน Building2 จะประกอบด้วยข้อมูลดังต่อไปนี้

- Id : string รหัสของตึกอาคาร
- name : string ชื่อตึกอาคาร
- lat : double ตำแหน่งละติจูด
- lon : double ตำแหน่งลองจิจูด

Placemark[] find_placemark(string key,string token)

ค้นหารายการสถานที่อื่นๆ โดยใช้ keyword

Paramter

key : string คำที่ใช้ค้นหา โดยสามารถใช้ key ได้หลายคำโดยใช้เว้นวรรคเป็นตัวขึ้น โดยจะใช้ operator AND ในการค้นหา

token : string token ที่ได้จากการยืนยันตัวตน

Return

Placemark[] Array รายการข้อมูลสถานที่อื่นๆที่ตรงกับ keyword ใน Placemark จะประกอบด้วยข้อมูลดังต่อไปนี้

- name : string ชื่อสถานที่
- latitude : double ตำแหน่งละติจูด
- longitude : double ตำแหน่งลองจิจูด

ฟังก์ชันสำหรับค้นหาข้อมูลโดยระบุ ตำแหน่ง Latitude , Longitude

Restaurant[] getAllRes(double latitude,double longitude,int distance,string token)

ค้นหารายการร้านอาหาร โดยใช้ ตำแหน่ง Latitude , Longitude

Paramter

เอกสารนี้เป็นเอกสารที่เขียนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Latitude,Longitude : double ตำแหน่งร้านอาหาร

Distance : int รัศมีในการค้นหาจากตำแหน่งที่ระบบ มีหน่วยเป็นเมตร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Restaurant[] Array รายการของร้านอาหารในบริเวณที่ระบุ

Building2[] getAllRes(double latitude,double longitude,int distance,string token)

ค้นหารายการตึกอาคาร โดยใช้ ตำแหน่ง Latitude , Longitude

Parameter

Latitude,Longitude : double ตำแหน่งตึกอาคาร

Distance : int รัศมีในการค้นหาจากตำแหน่งที่ระบบ มีหน่วยเป็นเมตร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Building2[] Array รายการของตึกอาคารในบริเวณที่ระบุ

Placemark[] getAllRes(double latitude,double longitude,int distance,string token)

ค้นหารายการสถานที่อื่นๆ โดยใช้ ตำแหน่ง Latitude , Longitude

Parameter

Latitude,Longitude : double ตำแหน่งสถานที่อื่นๆ

Distance : int รัศมีในการค้นหาจากตำแหน่งที่ระบบ มีหน่วยเป็นเมตร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Building2[] Array รายการของสถานที่อื่นในบริเวณที่ระบุ

ฟังก์ชันอื่นๆ

Restaurant getrestaurant(string rid,string token)

เรียกข้อมูลร้านอาหาร โดยระบบ หมายเลข id ร้านอาหาร

Parameter

rid : string รหัสหมายเลขร้านอาหาร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Restaurant ข้อมูลของร้านอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RMap[] getpicture(string rid,string token)

เรียกรูปภาพประกอบร้านอาหาร โดยระบบ หมายเลข id ร้านอาหาร

Paramter

rid : string รหัสหมายเลขร้านอาหาร

token : string token ที่ได้จากการยืนยันตัวตน

Return

RMap[] array รวมรูปภาพร้านอาหาร ข้อมูลภายในมี

Mapdata : byte[] เป็นข้อมูลรูปภาพ

Building getbuilding(string id,string token)

เรียกข้อมูลตึกอาคาร โดยระบบ หมายเลข id

Paramter

id : string รหัสหมายเลขตึกอาคาร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Building ข้อมูลของตึกอาคารแบบสมบูรณ์ มีข้อมูลภายในดังนี้

- Id : string รหัสของตึกอาคาร
- name : string ชื่อตึกอาคาร
- lat : double ตำแหน่งละติจูด
- lon : double ตำแหน่งลองจิจูด
- buildingpicture : byte[][] รายการรูปภาพของตัวตึกอาคาร
- floor : floor[] array ข้อมูลของแต่ละชั้นภายในตึก มีข้อมูลดังนี้
 - background : byte[] ภาพแผนที่ในแต่ละชั้น
 - id : int หมายเลขชั้น
 - name : string ชื่อชั้น
 - polygon : polygon[] รายการพิกัดร้านอาหารภายในตึก มีข้อมูลภายในดังนี้
 - id : string หมายเลขร้าน
 - name : string ชื่อร้าน
 - desc : string คำอธิบายของร้าน
 - shopid : หมายเลขร้านอาหารอ้างอิง
 - shopname : ชื่อร้านอาหารอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- point : interiorpoint[] ข้อมูลพิกัดร้านอาหารระบุเป็นแบบ polygon

Reserve[] get_reserve(string token)

เรียกข้อมูลการจองร้านอาหารของสมาชิก

Paramter

token : string token ที่ได้จากการยืนยันตัวตน

Return

Reserve[] array ข้อมูลการจองภายใน ข้อมูลภายในมี

Restaurant : Restaurant ข้อมูลร้านอาหาร

Table : int หมายเลขโต๊ะที่จอง

Time,Endtime : string ช่วงเวลาที่จอง

Boolean insert_res(Restaurant res,string token)

เพิ่มข้อมูลร้านอาหาร โดยสมาชิกทั่วไป

Paramter

Res : Restaurant ข้อมูลของร้านอาหาร

token : string token ที่ได้จากการยืนยันตัวตน

Return

Boolean ผลลัพธ์การเพิ่มร้านอาหาร มีค่าเป็น

True : สำเร็จ

False : ไม่สำเร็จ

TmgData getmaptmg(string iduser,double lat,double lon)

ดึงภาพแผนที่จาก webservice โดยระบบตำแหน่งละติจูด และ ลองจิจูด

Paramter

Iduser : string รหัสสมาชิก

Lat,Lon : double ตำแหน่งแผนที่

Return

TmgData ข้อมูลแผนที่ มีข้อมูลภายในดังนี้

Mapdata : byte[] ข้อมูลรูปภาพ

Len : double ระยะเวลาที่มีความกว้างของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การติดต่อ Web service ภายในตัวโปรแกรม mobileREST

การติดต่อกับ web service ได้สร้างเป็น Project ขึ้นมาใหม่ 1 Project ชื่อ service สำหรับจัดการกับการเรียกใช้งานฟังก์ชันต่างๆของเว็บเซอร์วิสโดยเฉพาะกำหนด namespace ชื่อ service ประกอบด้วย class

CENTERSERVICE : คลาสสำหรับเรียกฟังก์ชันของเว็บเซอร์วิสกลาง

RESTSERVICE : คลาสสำหรับเรียกฟังก์ชันของเว็บเซอร์วิสร้านอาหาร

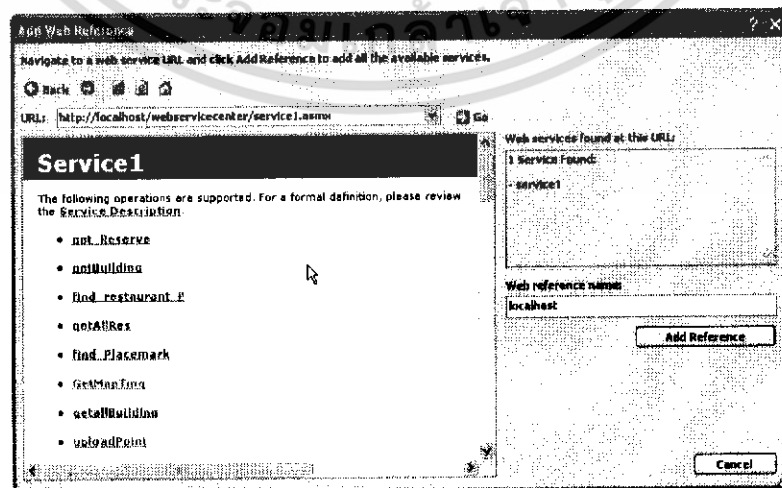


รูปที่ 3.7 Solution Explorer ในส่วนของ Service

การติดต่อ web service เพื่อดึงโครงสร้างของ function และ data type ที่ใช้เรียกงาน web service เพื่อให้ ตัว Project Service ที่สร้างขึ้นมารู้จักรูปแบบการเรียกใช้งาน web service ได้ถูกต้องตามที่ web service กำหนด

การติดต่อกับ webservice กลาง

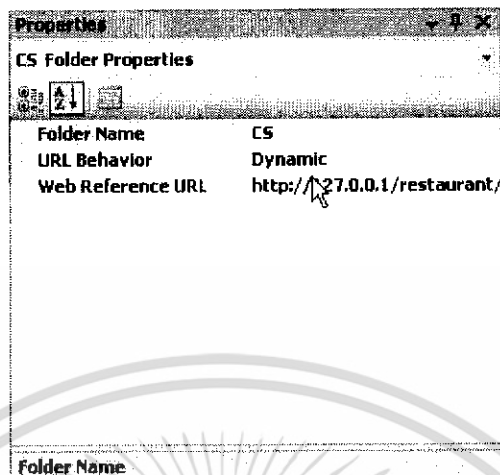
Add Web Reference Url : <http://localhost/webservicecenter/service1.asmx>



รูปที่ 3.8 การเพิ่ม Web Reference จาก Web service กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนด Webservice Properties : Folder Name : CS , Url Behavior : dynamic



รูปเข้า 3.9 Properties ของ Web service กลาง

การสร้าง Instance จาก class ของ service กลาง ที่ใช้ชื่อ Folder CS

```
public CS.Service1 cs = new CS.Service1();
```

กำหนด Url ให้ กับตัว Instance เพื่อเชื่อมต่อกับ webservice

```
cs.Url = url;
```

เรียกฟังก์ชัน สำหรับยืนยันตัวตน checkpwd โดยส่ง Username และ Password

```
public string token = "";
```

```
token = cs.checkPWD(username, password);
```

เรียกฟังก์ชัน GetProfile โดยส่ง Username และ Password เพื่อดึงข้อมูลประวัติส่วนตัว

```
public CS.GeneralUser gu;
```

```
gu = cs.GetProfile(username, password);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง Function ของ คลาส CenterService ที่สร้างขึ้นมาเรียกใช้งาน function ของ Web service
กลาง

```

public ArrayList Search(string key)
{
    if (token == null) return new ArrayList();

    ArrayList arr = new ArrayList();

    try
    {
        Service.CS.Restaurant[] r = cs.find_Restaurant(key, token);
        Service.CS.Building2[] b = cs.find_Building(key);
        Service.CS.Placemark[] p = cs.find_Placemark(key, token);
        if (r != null) for (int i = 0; i < r.GetLength(0); i++) arr.Add(r[i]);
        if (b != null) for (int i = 0; i < b.GetLength(0); i++) arr.Add(b[i]);
        if (p != null) for (int i = 0; i < p.GetLength(0); i++) arr.Add(p[i]);
    }
    catch
    {
    }

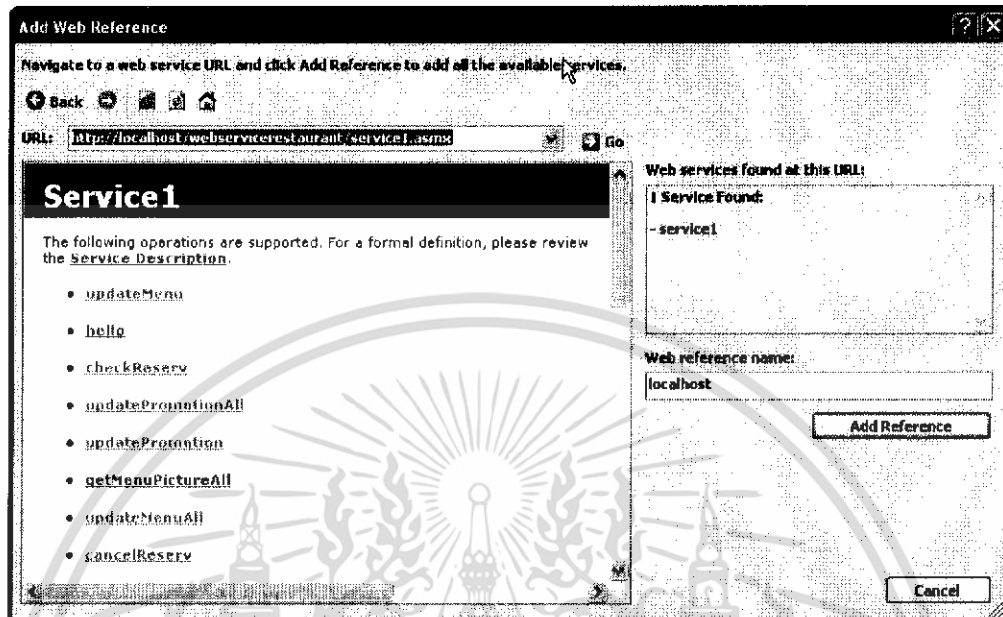
    return arr;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

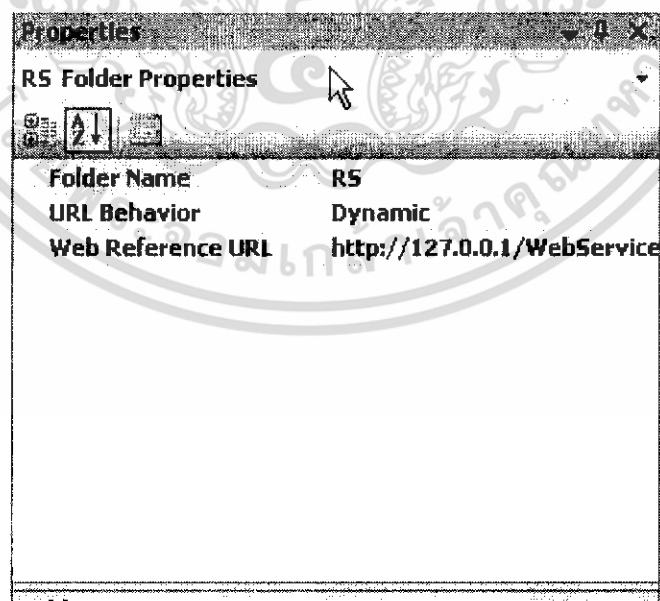
การติดต่อกับ webservice ร้านอาหาร

Add Web Reference Url : <http://localhost/websvicerestaurant/service1.asmx>



รูปที่ 3.10 การเพิ่ม Web Reference จาก Web service ร้านอาหาร

กำหนด Webservice Properties : Folder Name : RS , Url Behavior : dynamic



รูปที่ 3.11 Properties ของ Web service กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้าง Instance จาก class ของ service กลาง ที่ใช้ชื่อ Folder RS
 public RS.Service1 rs = new RS.Service1();

กำหนด Url ให้ กับตัว Instance เพื่อเชื่อมต่อกับ webservice
 rs.Url = url;

3.5 การติดต่อ GPS ภายในตัวโปรแกรม mobileREST

การติดต่อกับ GPS ได้สร้างเป็น Project ขึ้นมาใหม่ ชื่อ mobileGPS สำหรับจัดการกับการเรียกใช้งานฟังก์ชันต่างๆของ GeoFramework.Gps.PocketPC โดยเฉพาะกำหนด namespace ชื่อ mobileGPS ประกอบด้วย class

FGPS : คลาสสำหรับเรียกฟังก์ชันของ GPS



รูปที่ 3.12 Solution Explorer ในส่วนของ mobileGPS

การติดต่อ GPS เพื่อดึง class FGPS ซึ่งสืบทอดมาจาก Control Form เพื่อนำไปเป็น Attribute ของ form FSETTING ดังในรูป

<input type="radio"/> GPS Emulator	<input type="button" value="Search"/>
<input checked="" type="radio"/> Available Device	<input type="button" value="Start"/>
<input type="radio"/> Specific GPS Device	<input type="button" value="Stop"/>
Latitude	Longitude
Status.	

รูปที่ 3.13 หน้าตาของ control fGPS ที่สร้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ดูแลเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้าง Instance จาก class FGPS ของ mobileGPS

```
public mobileGPS.FGPS fGPS = new mobileGPS.FGPS();
```

การกำหนดตำแหน่งและคุณสมบัติของ Instance ลงใน form

```
fGPS.Location = new Point(10, 144);
fGPS.Width = 220;
fGPS.Height = 140;
Controls.Add(fGPS);
```

ฟังก์ชันที่ใช้ดึงค่า Latitude และ Longitude จาก fGPS ใน form setting มาใช้ใน form map

```
private void bGPS_Click(object sender, EventArgs e)
{
    double lat=0;
    double lon=0;
    lat = form.fSetting.fGPS.lat;
    lon = form.fSetting.fGPS.lon;
    setPosition(lat,lon);
}
```

Module ที่ใช้ใน fGPS

Reference มาจาก GeoFramework.Gps.PocketPC จากเว็บไซต์ <http://www.geoframeworks.com/> ซึ่งเป็น Control สำหรับ Compact Dotnet Framework

การเรียกใช้งาน GeoFramework.Gps.PocketPC ต้องกำหนด Interpreter เพื่อรองรับ Event ของ NMEA protocol โดยจะประกาศ Interpreter ดังนี้

```
private GeoFramework.Gps.Nmea.NmeaInterpreter Interpreter;
Interpreter = new GeoFramework.Gps.Nmea.NmeaInterpreter();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Events ต่างๆ ของ Interpreter

void Interpreter_Disconnected(object sender,EventArgs e)

void Interpreter_Disconnecting(object sender,EventArgs e)

void Interpreter_Connected(object sender,EventArgs e)

void Interpreter_Connecting(object sender,CancelEventArgs e)

void Interpreter_TimeoutOccurred(object sender, ExceptionEventArgs e)

void Interpreter_PositionChanged(object sender, GeoFramework.PositionEventArgs e)

การค้นหาอุปกรณ์ GPS

โหมดในการค้นหาอุปกรณ์มี 3 ทางเลือกคือ

1. ใช้ Emulator ภายใต้นตัวเอง
2. ค้นหาและเลือกอุปกรณ์ที่เหมาะสมอัตโนมัติ
3. ค้นหาและให้เลือกอุปกรณ์ด้วยตนเอง

ฟังก์ชันในการเริ่มค้นหา

```
GeoFramework.IO.Devices.Detect();
```

Events ต่างๆ ของการค้นหาตัวอุปกรณ์ GPS

```
private void Devices_DeviceDetectionStarted(object sender, EventArgs e)
```

```
private void Devices_DeviceFailure(object sender,
```

```
GeoFramework.IO.DeviceEventArgs e)
```

```
private void Devices_DeviceDetectionExceptionOccurred(object sender,
```

```
GeoFramework.ExceptionEventArgs e)
```

```
private void Devices_DeviceDetectionCompleted(object sender, EventArgs e)
```

```
private void Devices_DeviceDetectionAttempted(object sender,
```

```
GeoFramework.IO.DeviceEventArgs e)
```

```
private void Devices_DeviceDetected(object sender,
```

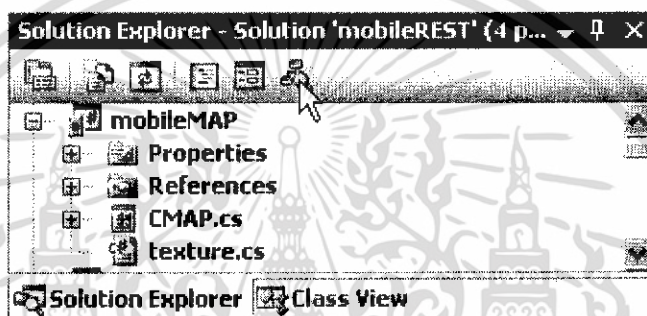
```
GeoFramework.IO.DeviceEventArgs e)
```

3.6 การติดต่อ MAP แบบ SETELLITE ของ GOOGLE ภายในตัวโปรแกรม mobileREST

การติดต่อกับ Map Google ได้สร้างเป็น Project ขึ้นมาใหม่ ชื่อ mobileMAP สำหรับจัดการกับการเรียกใช้งานฟังก์ชัน ในการแสดงผลภาพ Map โดยเฉพาะกำหนด namespace ชื่อ mobileMAP ประกอบด้วย class

FGPS : คลาส Control ที่แสดงผลภาพ Map Google

texture.cs : คลาสที่เป็นองค์ประกอบของภาพ Map ในแต่ละ Texture



รูปที่ 3.14 Solution Explorer ในส่วนของ mobileMAP

การสร้าง Instance จาก class CMAP ของ mobileMAP

```
mobileMAP.CMAP cMAP;
```

การกำหนดตำแหน่งและคุณสมบัติของ Instance ลงใน form

```
cMAP = new mobileMAP.CMAP();
```

```
cMAP.Location = new Point(10,30);
```

```
cMAP.Width = 220;
```

```
cMAP.Height = 220;
```

การดึงข้อมูลรูปภาพ satellite จาก URL ของ Google

- ระบุ ตำแหน่ง URL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เรียกฟังก์ชัน เพื่อคำนวณ เลขอ้างอิง ของ texture ในรูปของ ตัวเลข

```
public int lattoblocky(double lat)
```

```
public int lontoblockx(double lon)
```

- เรียกฟังก์ชัน เพื่อคำนวณหา หมายเลข server ที่บริการ texture จากเลขอ้างอิง ที่คำนวณออกมา

```
public int getserver(int blockx, int blocky)
```

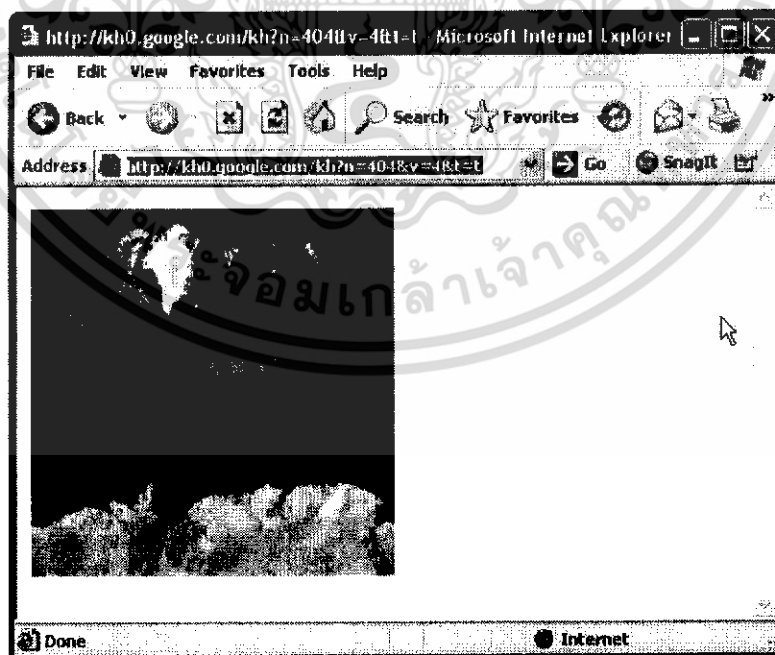
- เรียกฟังก์ชัน เพื่อคำนวณเข้ารหัสเลขอ้างอิงให้อยู่ในรูปของ URL

```
public string getkey(int blockx, int blocky)
```

การเข้ารหัสจากเลขอ้างอิง texture มาเป็น URL มีหลักการดังนี้

ค่าที่ได้จะมามีอยู่หลัง url “[http://kh0.google.com/kh?n=404&v=4&t=](http://kh0.google.com/kh?n=404&v=4&t=4)”

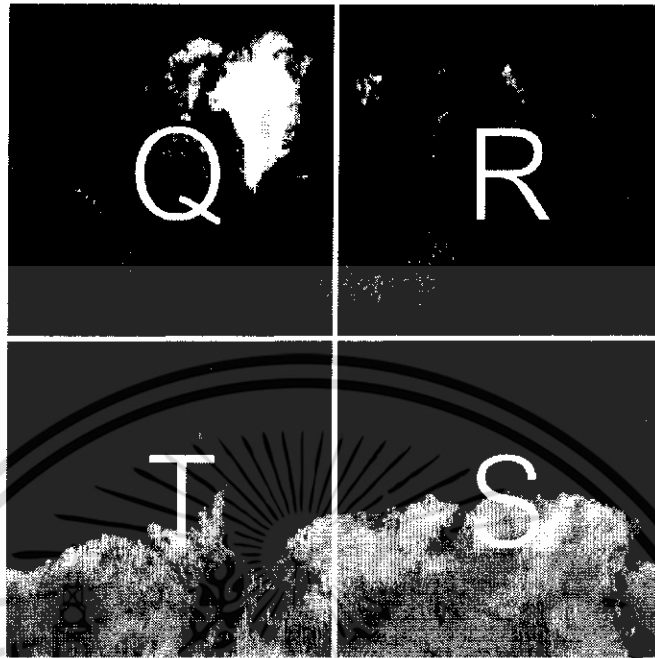
“4” คือ ภาพโดยรวมของ World Map



รูปที่ 3.15 ภาพที่ได้จาก url <http://kh0.google.com/kh?n=404&v=4&t=t>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในภาพจะแบ่งออกเป็น 4 ส่วน ดังนี้ เมื่อต้องการเข้าไปดูในสร้างใดส่วนหนึ่งใน 4 ส่วน ก็ให้เลือกตัวอักษรที่ต้องการระบุต่อไปใน URL



รูปที่ 3.16 แสดงลักษณะการแบ่งส่วนของ texture เพื่อ Zoom เข้าไปดูรายละเอียดภายในส่วนย่อย

ถ้าต้องการ แสดงรูปภาพเฉพาะในส่วนของ “r” ก็ค้ด้านขวาบนของรูปภาพเดิม ให้ใส่ “r” ต่อท้ายจาก url เดิมที่มีจะ ได้ผลลัพธ์ดังรูป



รูปที่ 3.17 ภาพที่ได้จากส่วนย่อย r จะได้ url <http://kh0.google.com/kh?n=404&v=4&t=tr>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 ภาพได้จากส่วนย่อย trsttqssrrqtqrrrrs จะได้ url
<http://kh0.google.com/kh?n=404&v=4&t=trsttqssrrqtqrrrrs>

ภาพที่ได้ จาก 1 url จะนับเป็น 1 texture เพื่อนำหลาย texture มาประกอบกับเป็นภาพใหญ่ จะต้องนำภาพที่มีหมายเลขอ้างอิง texture ที่ติดกันมากเข้ารหัส url และนำมาแสดงผลใน form โดยใช้ WebRequest เพื่อ ส่ง url ไปขอข้อมูลภาพจาก server ของ google ด้วยการเรียกใช้ ฟังก์ชันด้านล่าง โดยส่ง k คือ url เข้ารหัสมาจากเลขอ้างอิง texture และ slot คือหมายเลข server ที่ ถูกกำหนดไว้แล้ว

```
public void setkey(string k, int slot)
{
    if (key != k)
    {
        key = k;
        if (slot < 0) slot += 4;
        p.Image = null;
        p.Hide();
        wReq = WebRequest.Create("http://kh" + slot +
            ".google.com/kh?n=404&v=4&t=" + key);
        IAsyncResult r = (IAsyncResult)wReq.BeginGetResponse(new
            AsyncCallback(RespCallback), this);
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}

จาก sourcecode ด้านบน เป็นการเรียก WebRequest โดยทำงานโหมด Asynchronous โดยสร้าง Callback Function มารองรับการทำงานเมื่อได้รับการ Response กลับมา

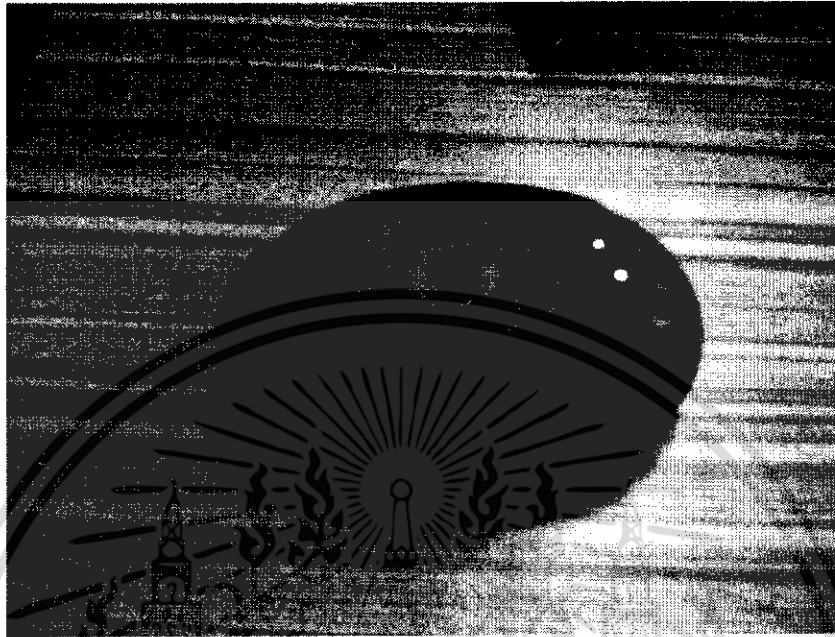
```
private void RespCallback(IAsyncResult r)
{
    texture t = (texture)r.AsyncState;
    try
    {
        t.wResp = (WebResponse)t.wReq.EndGetResponse(r);
        Stream data = t.wResp.GetResponseStream();
        t.p.Image = new Bitmap(data);
        data.Close();
        t.wResp.Close();
    }
    catch
    {
        t.wReq.Abort();
        t.key = "x";
    }
}
}
```

เมื่อได้รับ Response กลับมาให้รับข้อมูลที่ได้อีกมา สร้างเป็นภาพโดยอาศัย Class Bitmap แปลงข้อมูลเป็น Image เพื่อนำไปแสดงผลใน form ต่อไปโดยใช้ Picture Box ต่อ 1 texture

บทที่ 4

การทดลองและผลการทดลอง

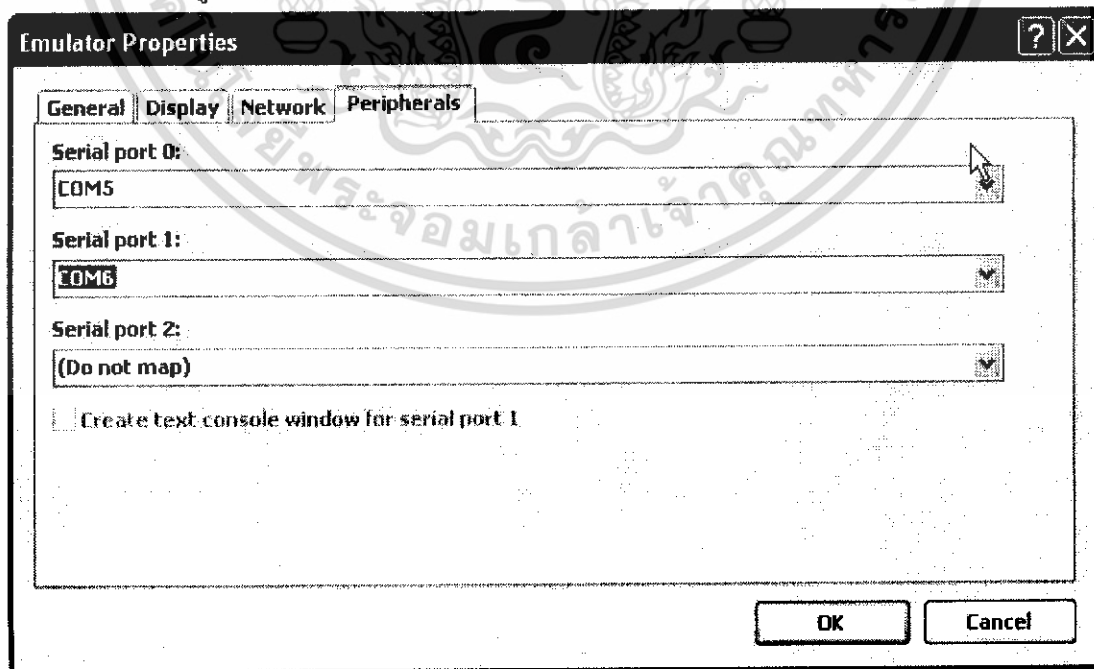
4.1 ขั้นตอนการ Config Device GPS



รูปที่ 4.1 ตัวอุปกรณ์ GPS ที่ใช้ทดลอง

Tap Menu File/Configure/Emulator Properties

จะได้หน้าต่างดังรูป

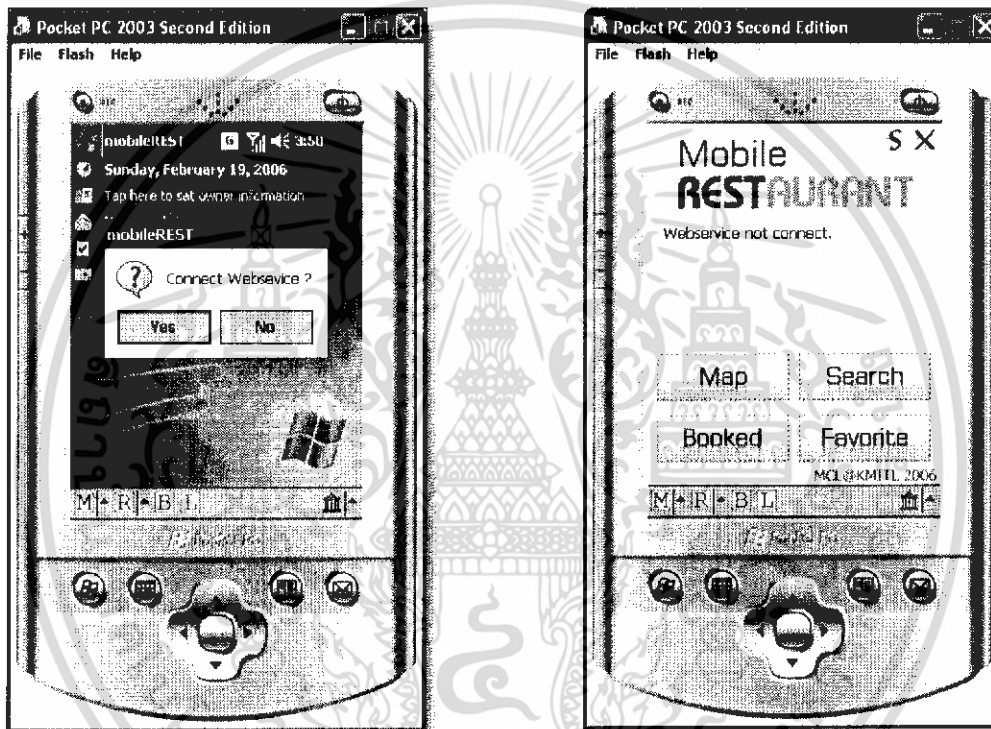


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.2 หน้าต่างกำหนด Serial port สำหรับ Emulator
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป ให้เลือก Serial port 0 เป็น Com portที่จะใช้งาน ส่วนตัวอย่างในรูป จะเลือกSerial com 5

4.2 แบบจำลองที่ใช้ในการทำงานของโปรแกรม บน Emulator

เมื่อเปิดโปรแกรม Mobile Restaurant ขึ้นมาที่แรก จะแสดงหน้าตาของโปรแกรมดังรูปที่ 4.3 โดยจะขึ้นเป็นหน้าต่างเพื่อยืนยันการเชื่อมต่อกับ web service หรือมี การทดลองนี้ให้ตอบว่า No คือไม่ต้องการเชื่อมต่อกับ web service ซึ่งเราจะไว้ config กันทีหลัง ซึ่งถ้าตอบว่า No แล้ว จะขึ้นหน้าต่างดังรูปที่ 4.4 ซึ่งรูปนี้แสดงหน้าต่างในขณะที่ไม่ได้เชื่อมต่อกับ web service โดยจะขึ้น คำว่า " Webservice not connect"



รูปที่ 4.3 หน้าต่างเพื่อยืนยันการเชื่อมต่อกับ web service รูปที่ 4.4 แสดงหน้าต่างในขณะที่ไม่ได้เชื่อมต่อกับ web service

การทดลอง โดยเข้าเมนูConfig GPS และ Webservices ที่ตัวS ด้านบนขวา จะปรากฏ หน้าจอดังรูปที่ 4.5 ซึ่งหน้าจอนี้จะแบ่งออกเป็น 2 โมดูล คือ โมดูลบนแสดงเป็น Textfield ให้ใส่ Username Password และ URL ของ webservices ที่จะทำการเชื่อมต่อ โดยเราจะป้อนข้อมูลดังนี้

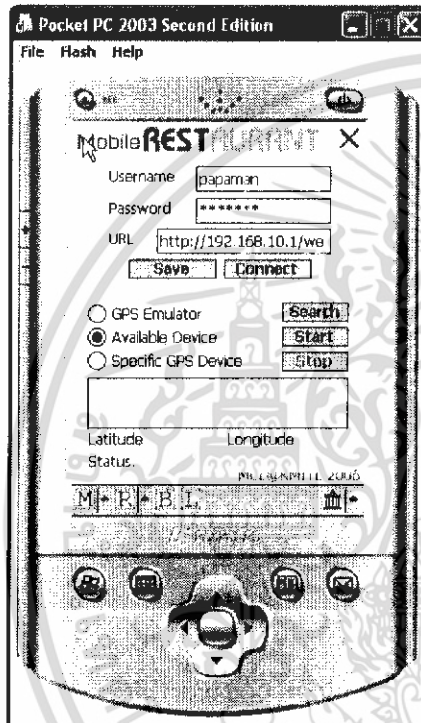
Username : papaman

Password : manpapa

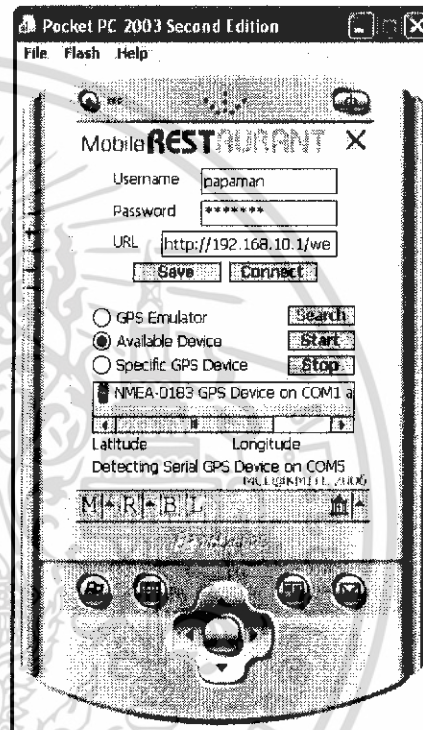
URL : <http://192.168.10.1/websevicecenter/service.asmx>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อป้อนข้อมูลดังกล่าวแล้ว ก็ให้กดปุ่ม Connect หรือต้องการ Save ข้อมูลเก็บไว้ก็ กดที่ปุ่ม Save ส่วนโมดูลต่าง เป็นโมดูลเกี่ยวกับการ Config GPS โดยแท้ GPS Emulator เป็นแท้เมนูไว้เลือกเชื่อมต่อ GPS แบบการจำลอง โดยเราจะทำการทดลองเลือก Available Device เพื่อให้ทำการค้นหา Com Port ติดต่อ GPS เอง อัตโนมัติ ส่วนแท้ Specific GPS Device เป็นการกำหนด Com Port ที่เราเลือกเอง ซึ่งเราเลือก Com5 ไว้ จะปรากฏ อุปกรณ์ NMEA-0183 GPS Device ดังรูปที่ 4.5 เมื่อทำการการ Config แล้วจะเข้าสู่หน้าต่างหน้าแรกของตัวโปรแกรม mobileREST ดังรูปที่ 4.7 ซึ่งจะแสดงรายละเอียด " Webservice connected "



รูปที่ 4.5 หน้าต่าง setting เพื่อกำหนด username password และเชื่อมต่อ GPS Receiver



รูปที่ 4.6 หน้าแสดงผลลัพธ์การค้นหาอุปกรณ์ GPS Receiver

ส่วนด้านล่างจะปรากฏเมนู หลัก 4 เมนู คือ เมนู Map เมนู Search เมนู Booked และ เมนู Favorite

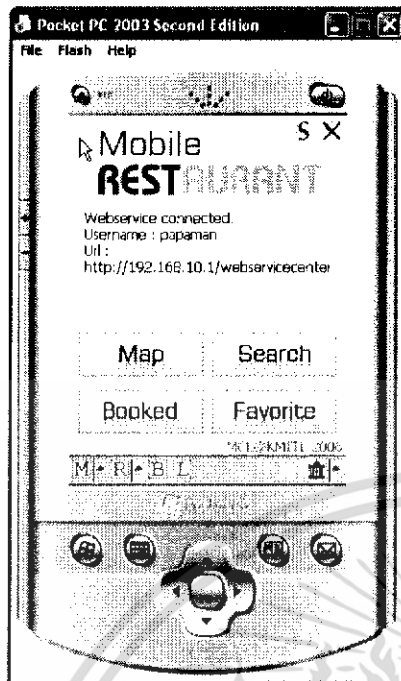
เมนู Map แสดงหน้าจอเกี่ยวกับแผนที่

เมนู Search แสดง

เมนู Booked

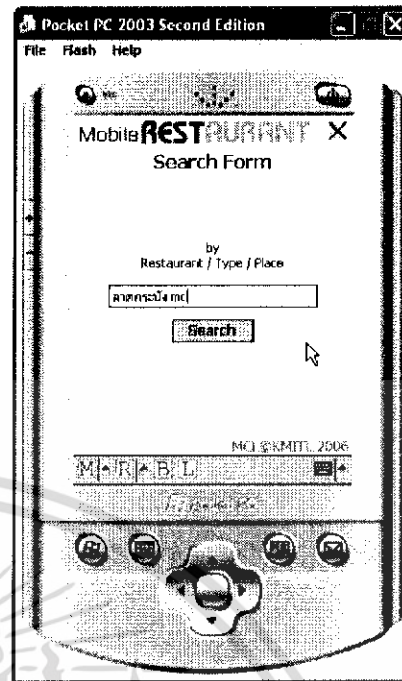
เมนู Favorite

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



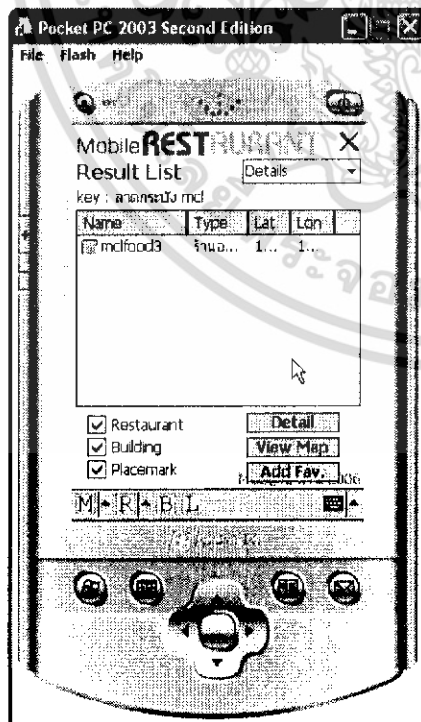
รูปที่ 4.7 หน้าแรกของตัวโปรแกรม

mobileREST



รูปที่ 4.8 หน้าการค้นหา

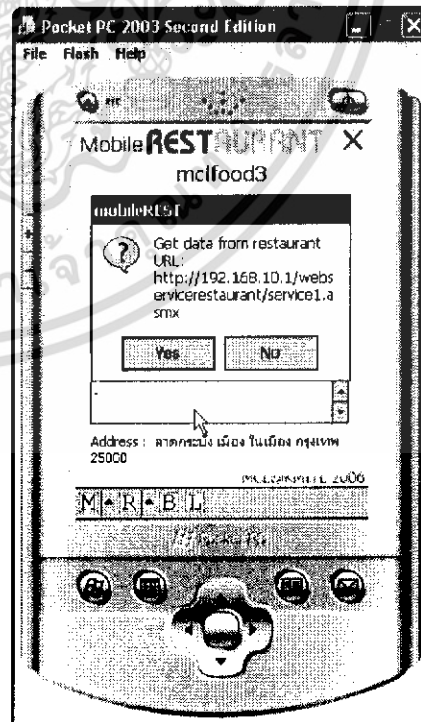
ส่วนรูปที่ 4.8 เป็นหน้าแสดงการค้นหาร้านอาหาร โดยแสดง input เป็น ชื่อร้าน ย่านหรือสถานที่ เป็นต้น โดยเราจะป้อน input ที่ต้องการจะค้นหาว่า “ลาดกระบัง mcl” ผลลัพธ์ ที่ได้แสดงดังรูปที่ 4.9 ซึ่งจะแสดงร้าน mclfood3 ส่วน รูปที่ 4.10 จะแสดงหน้าต่างการเชื่อมต่อ web service ของร้านอาหาร mclfood3



รูปที่ 4.9 แสดงรายการร้านอาหารที่ได้จากการ

เอกสารเป็นเอกสารที่ส่งมาไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านการค้า

ค้นหาด้วยคำว่า ลาดกระบัง mcl



รูปที่ 4.10 แสดงหน้าต่างการเชื่อมต่อ web

service ของร้านอาหาร

ไม่ว่ากรณีใดๆ ฟงสน อีกทงห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

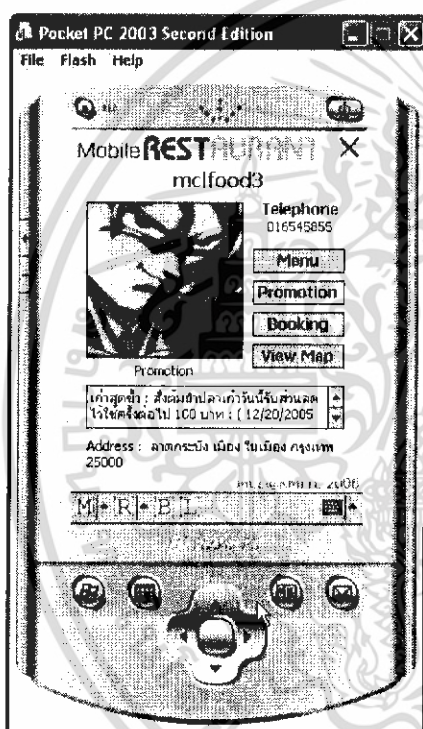
ต่อมา จะแสดงหน้าแรกของร้านอาหาร mclfood3 ที่ทำการทดลอง ดังรูปที่ 4.11 โดยจะแสดงรายละเอียดของร้านอาหาร เช่นที่อยู่ของร้าน mclfood3 เบอร์โทรศัพท์ รายละเอียดของ Promotion โดยหน้าต่างนี้จะแสดงบรรยากาศของร้าน mclfood3 ด้วย ซึ่งดังรูปจะแสดงเป็นรูปตัวการ์ตูนแทน แล้วมีแท็บเมนู ของร้านอาหารเพื่อเลือกดูรายละเอียด ทั้งหมด 4 เมนูดังนี้

แท็บเมนู Menu ไว้แสดงเมนูทั้งหมดของร้านอาหาร

แท็บเมนู Promotion ไว้แสดงรายละเอียดของPromotionของร้านอาหาร

แท็บเมนู Booking ไว้แสดงการจองโต๊ะล่วงหน้า

แท็บเมนู Viewmap ไว้แสดงแผนที่ของร้านอาหาร

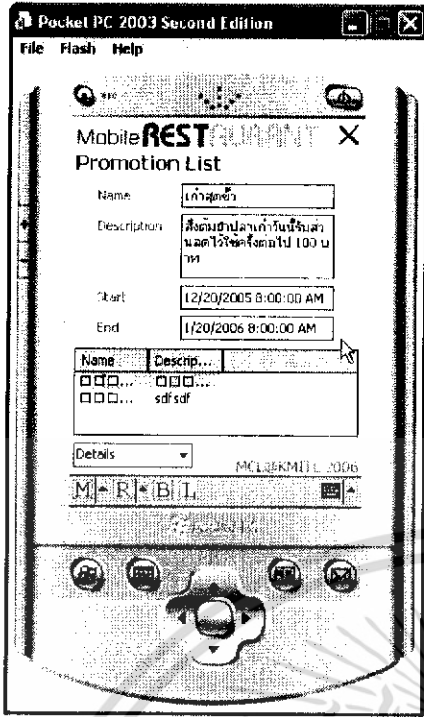


รูปที่ 4.11 หน้าแรกของร้านอาหาร

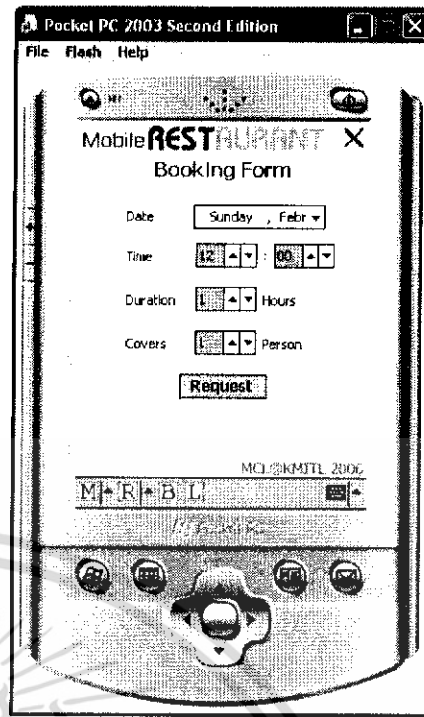


รูปที่ 4.12 แสดงรายการเมนูอาหารของร้าน

รายละเอียดของMenu List ของร้านอาหารจะแสดงเป็นรูปของอาหาร ซึ่งจะบอกชื่อของอาหาร ชนิด และราคาของอาหารนั้นๆ ดังรูปที่ 4.12 ส่วนรูปที่ 4.13 เป็นการแสดงหน้าของโปรแกรมชั้นของร้านอาหารที่ต้องการเลือก ซึ่งจะแสดงรายละเอียดชื่อเมนู รายละเอียดโปรแกรมชั้น ระยะเวลาของโปรแกรมชั้น

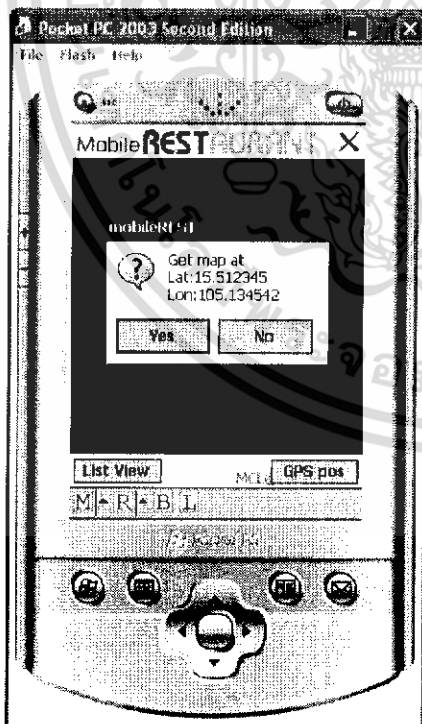


รูปที่ 4.13 หน้าโปรโมชัน ของร้านอาหาร

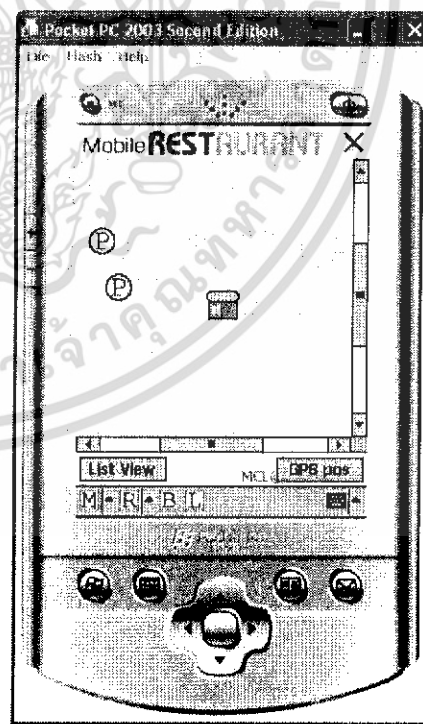


รูปที่ 4.14 หน้าต่างสำหรับการจองโต๊ะ

ร้านอาหาร
 การจองโต๊ะล่วงหน้าสามารถจองได้ที่เมนู Booking ซึ่งจะแสดงวัน เวลาการจอง
 ระยะเวลาที่จองและ จะจองล่วงหน้า ก็ โต๊ะ ดังรูปที่ 4.14



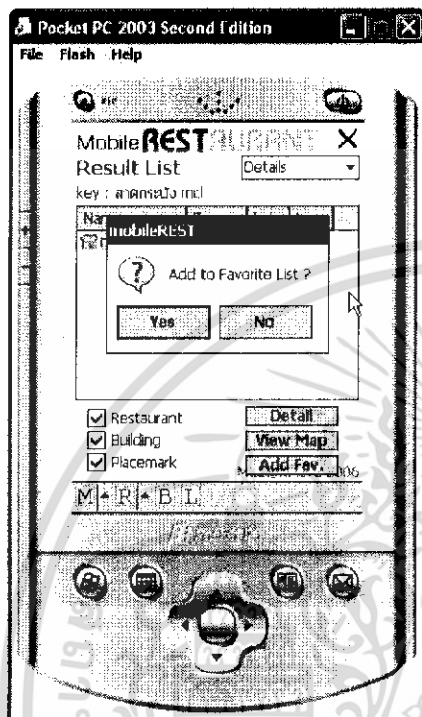
รูปที่ 4.15 แสดงหน้าต่างการเชื่อมต่อ map service เพื่อดึงภาพแผนที่



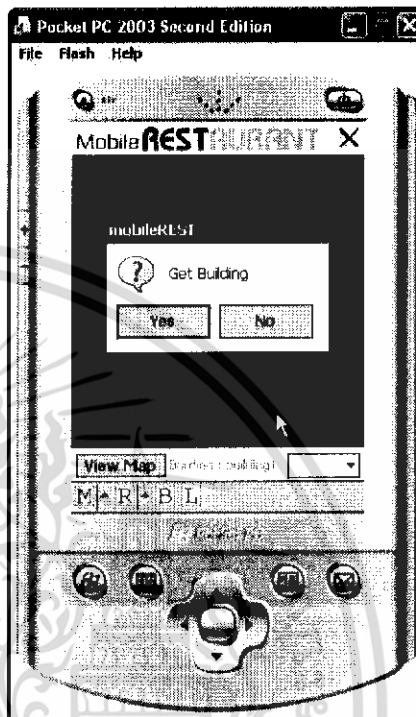
รูปที่ 4.16 แผนที่ในบริเวณที่ไม่มีข้อมูลจาก map service จะแสดงข้อมูลได้ที่เฉพาะของ service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้
 service ร้านอาหาร ระเบียบด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าต้องการจะ View Map เพื่อที่จะแสดงแผนที่ ก็เข้ามาแสดงที่เมนู View Map จะปรากฏหน้าต่างการเชื่อมต่อ map service เพื่อดึงภาพแผนที่ จาก ละติจูด และ ลองจิจูด ของร้านอาหาร mclfood3 ดังรูปที่ 4.15 ถ้าตอบ Yes จะแสดงหน้าต่างดังรูปที่ 4.16 เป็นร้านอาหาร mclfood3 เป็น รูปสี่ชมพู

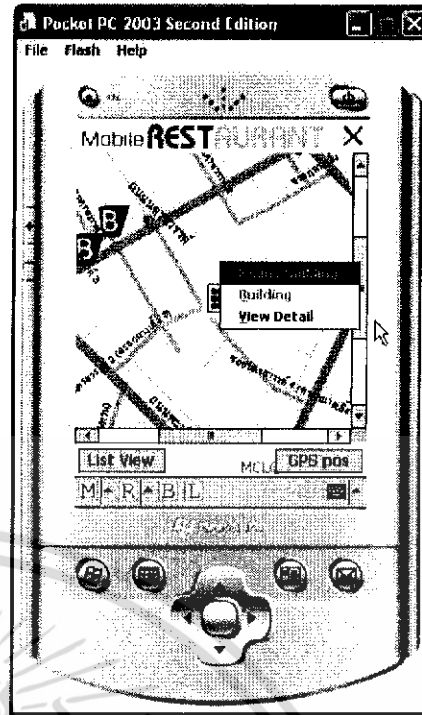
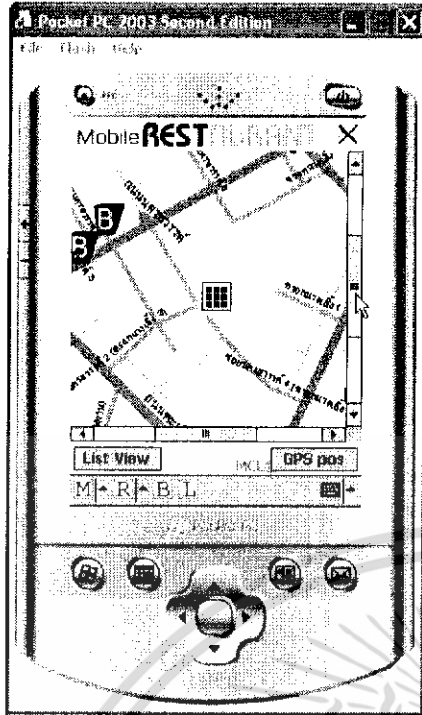


รูปที่ 4.17 แสดงการบันทึกข้อมูลลง Favorite



รูปที่ 4.18 แสดงหน้าการเชื่อมต่อ web service เพื่อดึงข้อมูลแผนที่ ภายในตึกอาคาร

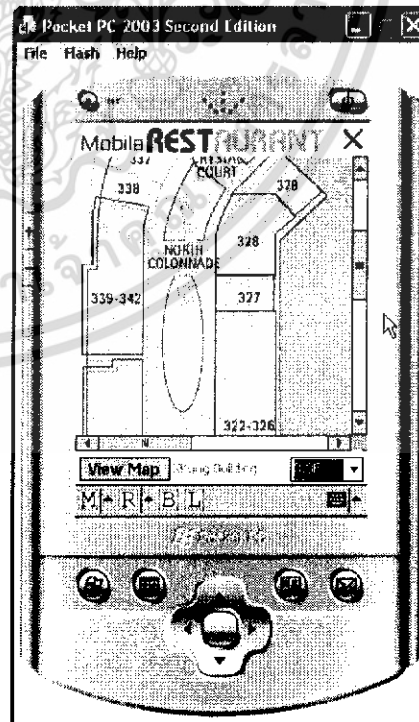
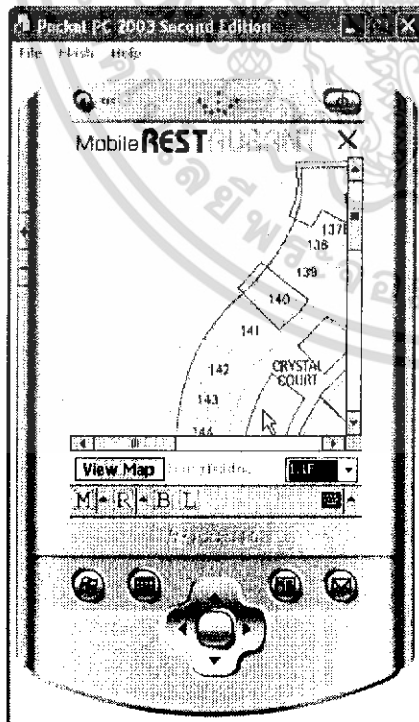
ส่วนรูปที่ 4.17 จะแสดงหน้าต่างการบันทึกร้านอาหารลง Favorite ถ้าตอบ yes คือ Add ส่วนเลือก No คือ ไม่Add ส่วนการทดลอง ที่จะแสดงหน้าการเชื่อมต่อ web service เพื่อดึงข้อมูลแผนที่ ภายในตึกอาคาร ดังรูปที่ 4.18 เพื่อดูแผนที่ภายในตึกของร้านอาหารนั้นอยู่ ซึ่งจะแสดงดังรูปที่ 4.19 จะแสดงแผนที่ ที่มี ตึกอาคาร ซึ่งสามารถคลิกเข้าไปดูร้านอาหารภายในตึกได้โดยแยกเป็นชั้น แล้วสามารถเข้าเพื่อดู View Detail ดังรูปที่ 4.20



รูปที่ 4.19 แผนที่ ที่มี ตึกอาคาร ซึ่งสามารถคลิก
เข้าไปดูร้านอาหารหายในตึกได้โดยแยกเป็น
ชั้น

รูปที่ 4.20 แสดงตึกอาคาร ซึ่งสามารถคลิกเข้า
ไปดูร้านอาหารหายในตึกได้โดยแยกเป็นชั้น

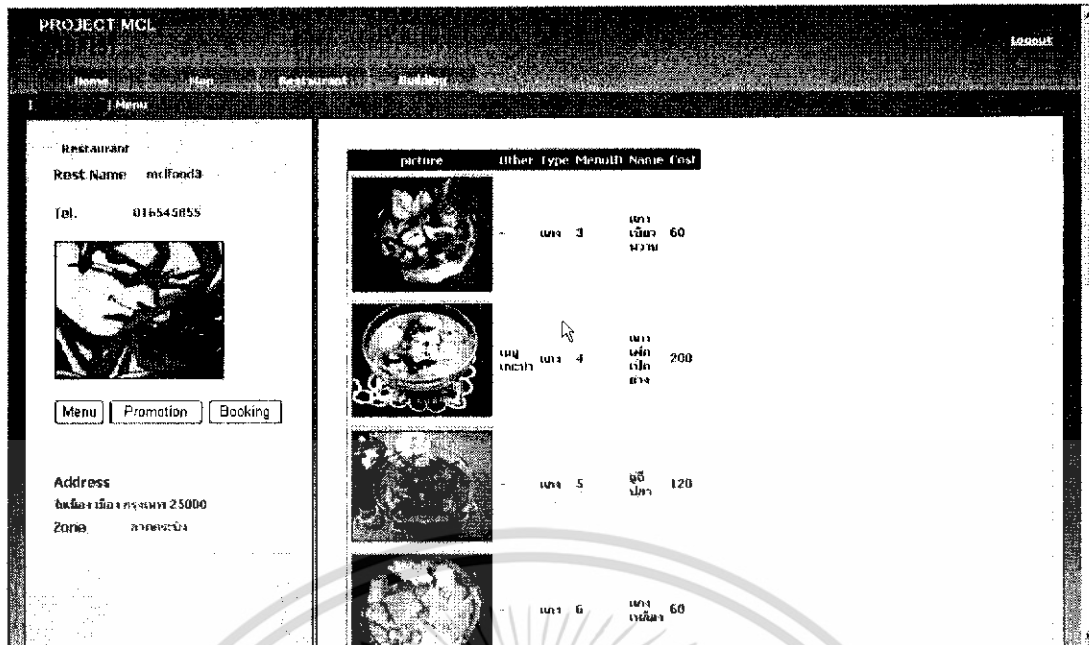
ผลลัพธ์ของการดูแผนที่ ภายในร้านอาหาร ดังรูปที่ 4.21 และ รูปที่ 4.22



รูปที่ 4.21 ภาพแผนที่ภายใน ตึก อาคาร

รูปที่ 4.22 ภาพแผนที่ภายใน ตึก อาคารอีกชั้น

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 หน้าร้านอาหารบนเว็บไซต์

4.3 สมรรถนะของระบบ

สามารถเรียกใช้งานฟังก์ชันพื้นฐานได้ครบ แต่ใช้เวลาในการประมวลผลนานเนื่องจากรันใน Emulator และจะต้องมีการเรียกใช้ Web Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทสรุป

จากการทดลองระบบโดยรวมแล้ว ระบบที่ได้พัฒนาขึ้นมาสามารถที่จะให้บริการเกี่ยวกับร้านอาหารได้อย่างถูกต้องตามความต้องการของผู้จัดทำ คือระบบสามารถให้บริการค้นหาร้านอาหาร โดยระบบจะแสดงตำแหน่งละติจูด, ลองจิจูดและแผนที่ปัจจุบัน สามารถตรวจสอบตำแหน่งและดูแผนที่ของร้านอาหารได้ โดยการอ้างตำแหน่งที่ต้องการ อาจจะเป็นจังหวัด, อำเภอ, ตำบล หรือค้นหาร้านอาหารบริเวณใกล้เคียง แสดงข้อมูลภายในร้านอาหาร ภาพถ่ายร้านอาหาร เรียกดูเมนู โปรโมชั่น ของร้านอาหาร สามารถบันทึกร้านอาหารที่ชื่นชอบได้ และจองที่นั่งของร้านอาหารได้อย่างถูกต้อง

5.2 ปัญหาที่เกิดขึ้น

1. ปัญหาเรื่องความเร็วในการรัน Emulator Pocket PC ที่ล่าช้า
2. ปัญหาเรื่องการ Update ฟังก์ชัน ของ Webservices ทำให้ต้องเสียเวลาในการปรับปรุงโปรแกรม ให้เข้ากับ Webservices
3. ปัญหาเรื่องการตอบสนองต่อการเรียกใช้ Webservices ที่ล่าช้า
4. ปัญหาเรื่องความเร็วของระบบเน็ตเวิร์คที่ล่าช้า

5.3 แนวทางการแก้ปัญหา

1. ทดสอบฟังก์ชันการทำงานบางอย่างบน Windows Application แล้วแปลง Code ลง Emulator
2. ทดลองโปรแกรมโดยการรันบน Windows Platform
3. ปัญหาเรื่องความล่าช้าของระบบเน็ตเวิร์คเป็นเรื่องของเทคโนโลยีซึ่งในอนาคตสิ่งเหล่านี้จะหมดไป

5.4 แนวทางการพัฒนาต่อ

เนื่องจากระบบนี้มีการทำงานอยู่ในวงจำกัดในเรื่องการค้นหาร้านอาหารเท่านั้น ในอนาคตผู้ใช้อาจต้องการข้อมูลที่มากกว่า ระบบนี้สามารถนำไปประยุกต์หรือเพิ่มเติมในส่วนอื่นๆ อาทิเช่นการค้นหาข้อมูลสถานที่สำคัญต่างๆในบริเวณที่ต้องการ เช่น ศาลา, สถานที่ท่องเที่ยว หรืออาจจะลงลึกไปถึงข้อมูลบุคคลทั่วไปซึ่งข้อมูลทั้งหมดมีทั้งละติจูดและลองจิจูดที่จะสามารถนำไประบุตำแหน่งในแผนที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] ศุภชัย สมพานิช., 2546, “คู่มือการเขียน โปรแกรมและการทำงาน Visual c# .NET ฉบับสมบูรณ์” , สำนักพิมพ์อินโฟเพรส , นนทบุรี.
- [2] มณีโชติ สมานไทย., 2546, “การเขียน โค้ด ASP.NET ฉบับสมบูรณ์” , สำนักพิมพ์อินโฟเพรส, นนทบุรี.
- [3] ยิ่งยศ สันติประเสริฐ, 2545, “เรียนรู้ ASP.NET ด้วย C#”, ซีเอ็ดยูเคชั่น , กรุงเทพฯ.
- [4] Microsoft Corporation, “MSDN”, [Online] URL : <http://msdn1.microsoft.com/en-us/default.aspx>
- [5] Microsoft Corporation, “ASP.NET”, [Online] URL : <http://www.asp.net/default.aspx>
- [6] GeoFrameworks, “GPS.NET for Pocket PC Compact Framework 2.0” , [Online] URL : <http://www.geoframeworks.com/>
- [7] GPS, “แหล่งข้อมูล GPS” [Online] URL : <http://www.falcom.de/>
- [8] GPS, “แหล่งข้อมูล GPS” [Online] URL : <http://www.kh-gps.de/index.htm>
- [9] GPS, “แหล่งข้อมูล GPS” [Online] URL : <http://www.gpsinformation.org/dale/nmca.htm>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้