

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

การพัฒนาเว็บไซต์เพื่อการเลือกซื้ออสังหาริมทรัพย์

WEB SERVICE DEVELOPMENT FOR REAL-ESTATE
PURCHASING



ฉพ.

๔/๑๕๒ ๗

๒๕๔๙

อาจารย์ที่ปรึกษา

ผศ.ดร. จันทร์บุรณ สติตวิริยวงศ์

เลขหมู่.....

เลขทะเบียน.....04427

วัน,เดือน,ปี - 5 ส.ย. 2551



H004427

b.....11๙ 22382
i.....

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอยู่ภายใต้เงื่อนไขของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคฤดูร้อน ปีการศึกษา 2549

**WEB SERVICE DEVELOPMENT FOR REAL-ESTATE
PURCHASING**



**A SYSTEM DEVELOPMENT PROJECT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECNOLOGY**

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
SUMMER/ 2006

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2007

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังฆาริมทรัพย์
นักศึกษา	นายมนตรี ขาววิชัย
รหัสนักศึกษา	47066427
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	ผศ.ดร.จันทร์บูรณ์ สถิตวิริยวงศ์

บทคัดย่อ

การพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังฆาริมทรัพย์ มีจุดประสงค์เพื่อแก้ปัญหา การหาข้อมูลบ้านจากหลายแห่ง เพื่อการตัดสินใจ โดยระบบที่พัฒนาจะเป็นเว็บกลางในการเรียกใช้เว็บเซอร์วิสเพื่อค้นหารายการคอนโดมีเนียม บ้าน ทาวเฮาส์ แบบบ้าน พร้อมทั้งโปรโมชั่นต่างๆในเว็บเดียว และเมื่อผู้ใช้เลือกบ้านได้แล้วก็ทำการส่งข้อมูลไปยังเว็บเซอร์วิสของบริษัทเจ้าของโครงการเพื่อให้ผู้ใช้ ติดต่อกับเจ้าของโครงการโดยตรง ในการพัฒนาระบบจะใช้เทคโนโลยี .NET และ JAVA เพื่อแสดงให้เห็นว่า เว็บเซอร์วิส สามารถทำงานข้ามแพลตฟอร์มได้

Title	Web Service Development for Real-estate Purchasing
Student	Mr. Montree Yawichai
Student ID.	47066427
Degree	Master of Science
Programme	Information Science
Academic Year	2006
Advisor	Asst.Prof. Dr.Chanboon Sathitwiriyawong

ABSTRACT

The purpose of web service development for real-estate purchasing is for solving the problem of project searching from many sources to make decision. The developed system will be the central web service to call the services for searching condominium projects, house projects, townhouse projects, models and promotions. When a customer decides to reserve the project, the system will send the reservation data to web service which owns the project. In this system development project, will use .net and java technology to demonstrate ability of web service in crossing platform.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. จันทรบูรณ์ สถิตวิริยวงศ์ ซึ่งเป็นอาจารย์ผู้ควบคุมโครงการพัฒนาระบบงาน ข้าพเจ้ารู้สึกซาบซึ้งในความอนุเคราะห์จากท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า ขอขอบคุณเพื่อนๆ พี่ๆ IS 18.1 ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะเทคโนโลยีสารสนเทศที่ให้ความช่วยเหลือ ในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำโครงการพัฒนาระบบงานฉบับนี้สำเร็จ ลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากโครงการพัฒนาระบบงานฉบับนี้ ข้าพเจ้าขอบแต่ผู้มีพระคุณทุกท่าน

มนตรี ยาวิชัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VIII
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมุติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในโครงการพิเศษ	2
1.5 การเปรียบเทียบระหว่างวิธีที่นำเสนอกับวิธีการแบบพื้นฐาน.....	3
1.6 ขอบเขตของโครงการพิเศษ	3
1.7 ขั้นตอนการศึกษา.....	3
บทที่ 2 ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง	
2.1 ทฤษฎีเว็บเซอร์วิส (Web Services)	4
2.2 XML	9
2.3 เทคโนโลยี.NET	11
2.4 Apache Axis.....	13
2.5 Google maps API.....	17
บทที่ 3 การวิเคราะห์และออกแบบระบบ	
3.1 การออกแบบเว็บทำเพื่อการเลือกซื้ออสังหาริมทรัพย์	19
3.2 Real-estate1 Web Services และ Real-estate 2 Web Services.....	34
3.3 การออกแบบระบบเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2	34

สารบัญ(ต่อ)

หน้า

บทที่ 4 การออกแบบฐานข้อมูล

4.1 ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2	50
4.2 ฐานข้อมูลของเว็บกลาง	55

บทที่ 5 การพัฒนาระบบ

5.1 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ	68
5.2 รายละเอียดการทำงานของระบบ	69

บทที่ 6 บทสรุป

6.1 สรุปผลการออกแบบและพัฒนาระบบงาน	68
6.2 ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ	68
6.3 ข้อจำกัดของระบบที่ออกแบบและพัฒนาขึ้น	69
6.4 ปัญหาและอุปสรรคระหว่างการออกแบบและพัฒนาระบบงาน	69
6.5 ข้อเสนอแนะและแนวทางในการพัฒนาระบบเพิ่มเติม	69

บรรณานุกรม	70
------------------	----

ภาคผนวก ก. ขั้นตอนการสมัครขอใช้บริการ Google Maps API	71
---	----

สารบัญตาราง

ตารางที่	หน้า
3.1 รายละเอียดของยูสเคส Register	21
3.2 รายละเอียดของยูสเคส Edit profile	21
3.3 รายละเอียดของยูสเคส View property	22
3.4 รายละเอียดของยูสเคส View condominium	23
3.5 รายละเอียดของยูสเคส View house	23
3.6 รายละเอียดของยูสเคส View townhouse	24
3.7 รายละเอียดของยูสเคส request for reserve.....	24
3.8 รายละเอียดของยูสเคส validate account	25
3.9 รายละเอียดของยูสเคส getHouse	36
3.10 รายละเอียดของยูสเคส getHouse Detail	36
3.11 รายละเอียดของยูสเคส getHouseModel	37
3.12 รายละเอียดของยูสเคส getTownhouse.....	37
3.13 รายละเอียดของยูสเคส getTownhouseDetail	38
3.14 รายละเอียดของยูสเคส getTownhouseModel	38
3.15 รายละเอียดของยูสเคส getCondominium	39
3.16 รายละเอียดของยูสเคส getCondominiumDetail	39
3.17 รายละเอียดของยูสเคส getCondominiumModel.....	40
3.18 รายละเอียดของยูสเคส request for reserve	41
4.1 รายละเอียดฐานข้อมูลของตาราง Reservation.....	51
4.2 รายละเอียดฐานข้อมูลของตาราง Project	52
4.3 รายละเอียดฐานข้อมูลของตาราง Prop_Type	52
4.4 รายละเอียดฐานข้อมูลของตาราง Model	53
4.5 รายละเอียดฐานข้อมูลของตาราง Project_Model	53
4.6 รายละเอียดฐานข้อมูลของตาราง Condominium_Model	53
4.7 รายละเอียดฐานข้อมูลของตาราง House_Model	54
4.8 รายละเอียดฐานข้อมูลของตาราง Townhouse_Model.....	54
4.9 รายละเอียดฐานข้อมูลของตาราง project_status.....	55
4.10 รายละเอียดฐานข้อมูลของตาราง Customer	56
4.11 รายละเอียดฐานข้อมูลของตาราง Company	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง(ต่อ)

หน้า

4.12 รายละเอียดฐานข้อมูลของตาราง Reservation.....	57
---	----



สารบัญรูป

รูปที่	หน้า
2.1 แบบจำลอง web service.....	5
2.2 โครงสร้างของ SOAP	7
2.3 โมเดลของเทคโนโลยี .NET.....	12
2.4 โครงสร้างพื้นฐานของผู้ให้บริการบนพื้นฐานของ Axis.....	15
2.5 โครงสร้างพื้นฐานของผู้ร้องขอบริการบนพื้นฐานของ Axis.....	15
2.6 GUI ของ tcpmon	16
2.7 ตัวอย่างเว็บแอปพลิเคชันที่สร้างด้วย Google Map API.....	17
3.1 ภาพรวมของระบบเว็บเซอร์วิสเพื่อการส่งของอสังหาริมทรัพย์.....	18
3.2 ยูสเคสไดอะแกรมของเว็บกลาง.....	20
3.3 คลาสไดอะแกรมของเว็บกลาง.....	26
3.4 ซีเควนซ์ไดอะแกรมของ register	28
3.5 ซีเควนซ์ไดอะแกรมของ Edit profile	29
3.6 ซีเควนซ์ไดอะแกรมของ view condominium	30
3.7 ซีเควนซ์ไดอะแกรมของ view house	31
3.8 ซีเควนซ์ไดอะแกรมของ view townhouse	32
3.9 ซีเควนซ์ไดอะแกรมของ request for reserve	33
3.10 ซีเควนซ์ไดอะแกรมของ validate account	33
3.11 ยูสเคสไดอะแกรมของ Web Service System	35
3.12 คลาสไดอะแกรมของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์.....	42
3.13 ซีเควนซ์ไดอะแกรมของ getHouse	43
3.14 ซีเควนซ์ไดอะแกรมของ getHouseDetail.....	44
3.15 ซีเควนซ์ไดอะแกรมของ getHouseModel	45
3.16 ซีเควนซ์ไดอะแกรมของ getTownhouse	45
3.17 ซีเควนซ์ไดอะแกรมของ getTownhouseDetail	46
3.18 ซีเควนซ์ไดอะแกรมของ getTownhouseModel	47
3.19 ซีเควนซ์ไดอะแกรมของ getCondominium	47
3.20 ซีเควนซ์ไดอะแกรมของ getCondominiumDetail	48
3.21 ซีเควนซ์ไดอะแกรมของ getCondominiumModel	49
3.22 ซีเควนซ์ไดอะแกรมของ request for reserve	49

สารบัญรูป(ต่อ)

	หน้า
4.1 ER Diagram ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1และ2.....	50
4.2 ER Diagram ของเว็บกลาง	56
5.1 หน้าจอการสมัครสมาชิกของเว็บกลาง.....	60
5.2 หน้าจอล็อกอินเข้าสู่ระบบ	60
5.3 ข้อมูลการแจ้งความจำนงของลูกค้าเมื่อเข้าสู่ระบบ	61
5.4 หน้าจอการแก้ไขข้อมูลสมาชิก	61
5.5 หน้าจอการค้นหาตามประเภทของอสังหาริมทรัพย์.....	62
5.6 หน้าจอการค้นหาข้อมูลโดยการกำหนดเงื่อนไข.....	62
5.7 หน้าจอแสดงรายละเอียดของ โครงการที่ตรงตามเงื่อนไข.....	63
5.8 หน้าจอรายละเอียดของแต่ละ โครงการ	64
5.9 หน้าจอแผนที่โครงการ.....	65
5.10 หน้าจอแผนที่โครงการสามมิติ.....	65
5.11 หน้าจอรายละเอียดแบบอสังหาริมทรัพย์ที่มีใน โครงการ.....	66
5.12 หน้าจอการจองโครงการ	67

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในการเลือกซื้อทรัพย์สินจำพวกอสังหาริมทรัพย์เราต้องมีการศึกษารายละเอียดต่างๆ เกี่ยวกับอสังหาริมทรัพย์ ต้องทำการค้นหารายละเอียดของโครงการ และมีการเปรียบเทียบหลายๆ โครงการและหลายๆ บริษัทเพื่อให้สามารถหา บ้าน ทาวน์เฮาส์ คอนโดมิเนียม ตามที่ต้องการได้ ซึ่งปัจจุบันหลายๆ บริษัทได้ใช้ช่องทาง อินเทอร์เน็ต เพื่อประชาสัมพันธ์โครงการของตัวเอง โดยใช้ Web application แต่ก็มีปัญหาตรงที่ผู้ที่ต้องการซื้ออสังหาริมทรัพย์ต้องหาข้อมูลหลายๆ เว็บไซต์กว่าจะสามารถตัดสินใจเลือกซื้อได้ ดังนั้นเพื่อแก้ปัญหาดังกล่าวจึงต้องอาศัยเครื่องมือ ที่สามารถทำให้ผู้ที่ต้องการซื้ออสังหาริมทรัพย์สามารถเลือกดูรายละเอียดโครงการต่างๆ ในเวลาเดียวกันได้

ในปัจจุบันได้มีมาตรฐานที่ใช้ในการรับส่งข้อมูล เป็นมาตรฐานเดียวกัน คือ XML ซึ่งเป็นรูปแบบที่อธิบายถึงรายละเอียดของโครงสร้างและแบบข้อมูลเป็นภาษาหรือชุดคำสั่งที่ใช้ในการรับส่งข้อมูลบนเว็บ XML จะทำให้การจัดการข้อมูลหรือเรียกใช้ข้อมูลจากแอปพลิเคชันต่างๆ จะเข้าสู่มาตรฐานเดียวกันโดยไม่ขึ้นกับแพลตฟอร์ม (platform) และ XML ก็เป็นตัวผลักดันให้เกิดการประยุกต์ใช้ที่เรียกว่า เว็บเซอร์วิส (Web Service)

ด้วยเหตุนี้จึงได้ใช้เทคโนโลยีเว็บเซอร์วิสเข้ามาช่วยแก้ปัญหาต่างๆ ในการทำธุรกรรมทางพาณิชย์อิเล็กทรอนิกส์ เพราะเว็บเซอร์วิสจะทำให้เว็บไซต์ที่ละเว็บไซต์สามารถติดต่อสื่อสารกันได้โดยการส่งข้อมูลในรูปแบบ XML ทำให้แต่ละเว็บไซต์ที่ทำธุรกิจเดียวกันไม่จำเป็นต้องใช้เครื่องมือ ภาษา หรือระบบปฏิบัติการที่เหมือนกัน ก็สามารถทำการติดต่อสื่อสารกันได้ อีกทั้งการค้นหารายการ โครงการต่างๆ ทำได้ง่ายขึ้น เพราะมี องค์กรที่ทำหน้าที่เก็บรวบรวมข้อมูลต่างๆ เพื่อให้การค้นหาทำได้ง่ายมากขึ้น

ดังนั้นจึงได้เกิดแนวคิดที่จะพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้ออสังหาริมทรัพย์ เพื่อให้ลูกค้าที่ต้องการซื้อสินค้าจำพวกอสังหาริมทรัพย์ สามารถเรียกดูข้อมูลจากหลายๆ แหล่งได้โดยไม่ต้องสนใจว่า แอปพลิเคชัน เหล่านั้นจะสร้างมาจากสถาปัตยกรรม ภาษา และฐานข้อมูลที่แตกต่างกัน อีกทั้งยังเป็นการประหยัดเวลา เพราะการส่งซื้ออสังหาริมทรัพย์หรือการจองโครงการต่างๆ ต้องการรายละเอียดที่มาก เพราะเป็นสินค้าที่ราคาแพง และไม่ได้ซื้อบ่อยกันมากนัก

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

โครงการพัฒนาระบบงานฉบับนี้มุ่งหวังเพื่อศึกษาเทคโนโลยีเว็บเซอร์วิส เพื่อที่จะนำมาใช้ในการพัฒนาเว็บไซต์กลางที่ให้บริการค้นหารายละเอียดเกี่ยวกับธุรกิจทางด้านอสังหาริมทรัพย์ จากหลายๆเว็บไซต์ที่เป็นเจ้าของโครงการ โดยมีจุดประสงค์หลัก เพื่ออำนวยความสะดวกให้แก่ผู้ใช้ ทำให้ผู้ใช้บริการสามารถค้นหารายละเอียดต่างๆของโครงการแบบบ้าน ทาวน์เฮาส์ คอนโดมิเนียม ได้มากขึ้นและง่ายขึ้น เพราะเป็นสินค้าที่ราคาแพง และไม่ได้ซื้อบ่อยกันมากนัก จึงมีความจำเป็นที่จะต้องทราบข้อมูลรายละเอียดที่มาก

1.3 สมมติฐานของการศึกษา

จากเว็บไซต์อสังหาริมทรัพย์ที่มีอยู่ในปัจจุบันจะเป็นลักษณะของใครของมัน และข้อมูลที่เป็นเจ้าของโครงการอสังหาริมทรัพย์ต่างๆต้องการนำเสนอต่อลูกค้า จะมีลักษณะที่คล้ายๆกัน เช่น ราคาเริ่มต้น ที่ตั้ง เนื้อที่ สถานที่แวดล้อม แต่รูปแบบในการนำเสนอในแต่ละเว็บไซต์อาจจะต่างกันบ้าง ดังนั้น จึงเป็นไปได้ในการที่จะใช้เว็บเซอร์วิสในการเป็นเว็บกลางในการค้นหา ข้อมูลเหล่านั้น ซึ่งจะส่งผลให้ได้ประโยชน์ทั้ง 3 ฝ่ายคือ เจ้าของโครงการได้ให้ข้อมูลได้เข้าถึงกลุ่มลูกค้ามากขึ้น ลูกค้าก็สามารถหาข้อมูลที่ตรงกับความต้องการของตัวเองได้มากขึ้นจากการเปรียบเทียบ และเจ้าของเว็บกลางสามารถมีรายได้จากค่าโฆษณา หรือจากเจ้าของโครงการจากที่ทำให้ลูกค้ามาจองโครงการได้มากขึ้น

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในโครงการพัฒนาระบบงาน

การใช้เว็บเซอร์วิสในธุรกิจอสังหาริมทรัพย์ทำได้ โดยเริ่มแรกเจ้าของเว็บไซต์ที่มีเว็บเป็นของตัวเองอยู่แล้วทำการเปิดบริการ เพื่อให้สามารถเรียกดูข้อมูลต่างๆของโครงการ หรือทำการจองได้ จากนั้นก็จะทำการประกาศบริการของตัวเอง จากนั้นเว็บกลางก็จะทำการเขียน โปรแกรมเรียกใช้บริการดังกล่าวตามที่ได้ระบุไว้ในไฟล์ wsdl สำหรับการค้นหาข้อมูลต่างๆจากเว็บไซต์นั้น มาแสดงผลให้ ผู้ใช้เว็บกลางทราบโดยการค้นหาจะสามารถ ค้นหาได้หลายรูปแบบ เช่นกำหนดเงื่อนไขว่าต้องราคาเริ่มต้นเท่าไร ประเภทของอสังหาริมทรัพย์ ที่ตั้ง รวมทั้งสามารถค้นหาโปรโมชั่นต่างๆของ แต่ละบริษัทที่เป็นเจ้าของโครงการ เมื่อจะทำการจองผู้ใช้บริการต้องทำการเข้าสู่ระบบเมื่อทำการจองข้อมูลการจองจะถูกส่งไปยังเว็บไซต์ที่เป็นเจ้าของโครงการนั้นๆ และจะทำการบันทึกไว้เป็นหลักฐานที่เว็บไซต์กลาง โดยในโครงการนี้จะทำการจำลองการทำงานโดยที่เว็บไซต์เจ้าของบริการจะพัฒนาด้วยภาษา JAVA และเว็บกลางจะใช้ .NET ซึ่งคิดว่าเมื่อนำไปใช้

เอกสารนี้จะทำให้เกิดความสะดวกแก่ทุกฝ่าย เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

การพัฒนาเว็บเซอร์วิสมาประยุกต์ใช้ในธุรกิจอสังหาริมทรัพย์ จะมีประโยชน์มากกว่าการที่แต่ละเว็บไซต์มีข้อมูลของตัวเอง แล้วให้ ผู้ใช้ทำการค้นหาเอง ซึ่งจะเป็นการสร้างความลำบากให้แก่ผู้ใช้ อีกทั้งผู้ใช้อาจจะพลาดข้อมูล โครงการที่ตรงกับความต้องการมากที่สุดก็ไม่ได้ เพราะการพาณิชย์ทางอสังหาริมทรัพย์ราคาแพงและไม่ค่อยซื้อหากันบ่อยนัก แม้ว่าปัจจุบันจะมีการใช้เว็บกลางเข้ามาช่วยแต่ก็ยังเป็นลักษณะของเว็บแอปพลิเคชัน ทำให้ผู้ทำการ update ต้องติดตามข่าวสารความเคลื่อนไหวอยู่ตลอดเพื่อให้ข้อมูลใหม่อยู่ตลอด ซึ่งเป็นการสร้างความลำบากแก่ผู้ทำการ update ดังนั้นเว็บเซอร์วิสจึงเป็นทางเลือกที่น่าสนใจมากกว่าวิธีดังกล่าว

1.6 ขอบเขตของโครงการพัฒนาระบบงาน

ในโครงการพัฒนาระบบนี้มีขอบเขตใช้ในประเทศไทย และปริมณฑล และจะทำการจำลองการทำงานของเว็บเซอร์วิสในความสามารถเป็นเว็บกลาง โดยที่ระบบสามารถค้นหาข้อมูลรายการโครงการในแต่ละประเภทได้ โดยที่ผู้ใช้ระบุเงื่อนไขในการค้นหา สามารถค้นหารายละเอียดของแบบบ้านที่ใช้ในแต่ละโครงการได้ ทำการสั่งจองโครงการได้ แต่ต้องเป็นสมาชิกของเว็บเท่านั้น โดยที่ระบบจะทำการส่งรายการจองไปยัง เว็บไซต์เจ้าของโครงการ และทำการเก็บไว้เป็นหลักฐานที่เว็บกลาง และผู้ใช้ที่เป็นสมาชิกสามารถดูรายการจองของตัวเองได้

1.7 ขั้นตอนของการศึกษา

โครงการพัฒนาระบบงานฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของโครงการ ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของโครงการ และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงการ โดยจะกล่าวถึงทฤษฎีของเว็บเซอร์วิส เทคโนโลยีต่างๆที่ใช้ในเว็บเซอร์วิส รวมถึงเทคโนโลยี .NET และ JAVA ที่ใช้ในเว็บเซอร์วิส

บทที่ 3 เป็นการวิเคราะห์และออกแบบระบบ โดยจะแสดงในรูปแบบของ Use Case Diagram, Class Diagram และ Sequence Diagram ตามลำดับ

บทที่ 4 เป็นการออกแบบฐานข้อมูลโดยใช้ ER Diagram

บทที่ 5 กล่าวถึงการติดตั้งและการทดสอบระบบ

บทที่ 6 เป็นบทสรุปผลโครงการและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและเทคโนโลยีที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการพัฒนาระบบงาน ความหมายของเว็บเซอร์วิสและเทคโนโลยีอื่นๆ ที่เกี่ยวข้องกับเทคโนโลยีเว็บเซอร์วิส

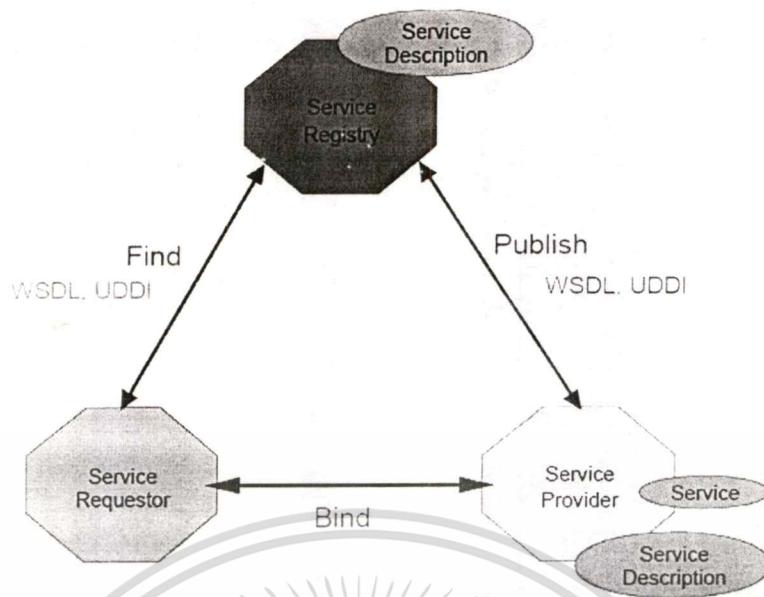
2.1 ทฤษฎีเว็บเซอร์วิส (Web Services)

2.1.1 นิยามของเว็บเซอร์วิส

เว็บเซอร์วิส คือ การติดต่อ (interface) ที่อธิบายถึง โอเปอเรชันต่างๆที่สามารถผ่านเครือข่ายโดยการผ่านมาตรฐานของข้อความ XML เว็บเซอร์วิสจะถูกอธิบายโดยใช้มาตรฐานรูปแบบของ XML ที่เรียกว่า รายละเอียดของบริการ(service description) ซึ่งมีรายละเอียดที่จำเป็นในการติดต่อกับบริการ รวมไปถึงรูปแบบข้อความ(คือรายละเอียดของโอเปอเรชัน) โปรโตคอลขนส่งและที่ตั้งของบริการ การติดต่อจะซ่อนรายละเอียดการติดตั้งของบริการ ส่งผลให้สามารถเป็นอิสระจากภาษาที่ใช้เขียนโปรแกรม ทำให้เว็บเซอร์วิส ขึ้นต่อกันน้อย (loosely coupled) เป็นเชิงคอมโพเนนท์ และติดตั้งข้ามเทคโนโลยีได้ เว็บเซอร์วิสทำให้งานเฉพาะหรือเซตของงานสมบูรณ์ขึ้น สามารถใช้แบบเดี่ยวๆหรือใช้ เว็บเซอร์วิสอื่นร่วมกันทำให้งานธุรกิจหรืองานที่ซับซ้อนสำเร็จลุล่วงไปได้ (Kregeri, H. 2001 : 6)

2.1.2 แบบจำลองเว็บเซอร์วิส

สถาปัตยกรรมของเว็บเซอร์วิส มีพื้นฐานอยู่ที่การติดต่อระหว่างกัน 3 บทบาท คือ ผู้ให้บริการ(service provider) แหล่งค้นหาบริการ(service registry) และผู้ขอใช้บริการ(service requestor) ในการติดต่อระหว่างกันจะประกอบไปด้วยโอเปอเรชันต่างๆ คือ การประกาศ การค้นหา และการผูก(bind) ซึ่งการกระทำเหล่านี้จะกระทำร่วมกันระหว่างสิ่งประดิษฐ์ของ web service หมายถึง ระหว่างโมดูลซอฟต์แวร์ของ เว็บเซอร์วิสและคำอธิบายของมัน ในบทบาททั่วไป ผู้ให้บริการจะกำหนดรายละเอียดของบริการสำหรับ เว็บเซอร์วิสและประกาศมันไปยังผู้ขอใช้บริการหรือแหล่งค้นหาบริการ ผู้ขอใช้บริการจะใช้โอเปอเรชันการค้นหาเพื่อค้นหาละเอียดของบริการจากที่ตนเอง หรือจากแหล่งค้นหาบริการ และใช้คำอธิบายบริการไปผูกกับผู้ให้บริการ และเรียกใช้หรือปฏิสัมพันธ์กัน องค์ประกอบของเว็บเซอร์วิสที่ประกอบไปด้วย ผู้ให้บริการ ผู้เรียกใช้บริการ และแหล่งค้นหาบริการแสดงดังรูปที่ 2.1



รูปที่ 2.1 แบบจำลองของเว็บเซอร์วิส

จากรูปที่ 2.1 จะมีการทำงานกันขององค์ประกอบทั้ง 3 บทบาทดังนี้ คือ

1. Service provider : ในมุมมองทางธุรกิจ ก็คือเจ้าของบริการนั่นเอง แต่ถ้าเป็นมุมมองทางด้านสถาปัตยกรรม ก็คือเฟล็ตฟอร์มที่โฮสต์ใช้เข้าถึงบริการ โดยสรุป service provider เป็นผู้ให้บริการ เป็นเครื่องserver ที่มีโปรแกรมการบริการเว็บเซอร์วิส ทำงานอยู่มีหน้าที่ในการเปิดบริการเพื่อรองรับการขอใช้บริการจาก requestor ที่เรียกเข้ามาขอใช้ ซึ่งรูปแบบของการให้บริการ อาจจะเป็นในเชิงธุรกิจ เช่น บริการรับใบสั่งซื้อสินค้าจากผู้ผลิตสินค้า บริการตรวจเครดิตลูกค้าในการอนุมัติสินเชื่อของธนาคาร หรืออาจจะเป็นการบริการทั่วไป เช่น บริการรายงานสภาพภูมิอากาศ

2. Service requestor : ในมุมมองทางธุรกิจ ก็คือ ธุรกิจที่ต้องการการทำงานบางอย่าง เพื่อให้ได้ผลลัพธ์อันน่าพอใจมาใช้ แต่ถ้าเป็นมุมมองทางด้านสถาปัตยกรรม ก็คือแอปพลิเคชันที่กำลังมองหาเพื่อทำการร้องขอ หรือเริ่มต้นปฏิสัมพันธ์กับบริการ โดยใช้ soap เป็นตัวกลาง

3. Service registry : เป็นแหล่งที่ใช้ค้นหาข้อมูลรายละเอียดของบริการว่าผู้ให้บริการที่มาประกาศรายละเอียดของบริการอยู่ที่ไหน Service registryจะทำหน้าที่เป็นตัวกลางให้กับผู้ให้บริการมาลงทะเบียนไว้ โดยใช้ไฟล์ WSDL บอกรายละเอียดของผู้ให้บริการและบริการที่มีให้ ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้ (Kregeri, H. 2001:7)

2.1.3 การปฏิบัติการในสถาปัตยกรรมเว็บเซอร์วิส

แอปพลิเคชันที่ต้องการใช้ประโยชน์จากเว็บเซอร์วิสจะต้องมี 3 พฤติกรรม คือ ประกาศ

รายละเอียดบริการ ค้นหารายละเอียดของบริการ และทำการเรียกใช้บริการตามที่ได้อธิบายไว้ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ทางการค้า
รายละเอียดบริการ พฤติกรรมเหล่านี้สามารถเกิดขึ้นเดี่ยวๆหรือซ้ำๆ รายละเอียดของพฤติกรรมทั้ง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 มีรายละเอียดดังนี้

1. การประกาศ (Publish) จำเป็นต้องประกาศรายละเอียดของบริการให้สาธารณรับทราบ เพื่อให้ผู้ที่ต้องการเรียกใช้บริการสามารถหามันได้
2. การค้นหา (Find) เป็นการกระทำที่ผู้ร้องขอบริการใช้ค้นหารายละเอียดของบริการโดยตรงหรือร้องขอจาก ที่เก็บบริการ (service registry) เพื่อให้ได้บริการที่ต้องการ
3. การผูก (Bind) ในการเรียกใช้บริการโอเปอเรชัน bind นี้เป็นการที่ผู้ร้องขอบริการกับผู้ใช้บริการใช้ในการติดต่อกันตามเงื่อนไขในรายละเอียดขอบริการ (Kregeri, H. 2001 : 7-8)

2.1.4 สิ่งประดิษฐ์ของเว็บเซอร์วิส (Artifacts of a Web Service)

สิ่งประดิษฐ์ของเว็บเซอร์วิส คือสิ่งสำคัญที่ผู้ใช้บริการของเว็บเซอร์วิสต้องการจะประกอบไปด้วย 2 สิ่ง คือ

1. บริการ (service) เป็นโมดูลของซอฟต์แวร์ที่ติดตั้งอยู่ที่ผู้ให้บริการ โดยที่ผู้ร้องขอบริการสามารถร้องขอใช้บริการผ่านเครือข่ายได้
2. รายละเอียดของบริการ (Service Description) จะประกอบไปด้วยรายละเอียดของการติดต่อและการติดตั้งของบริการ รวมไปถึงชนิดของข้อมูล การปฏิบัติการ การผูกข้อมูล (binding information) และที่ตั้งของบริการ รายละเอียดของบริการอาจจะประกอบไปด้วย metadata ที่อำนวยความสะดวกแก่ผู้ร้องขอบริการ รายละเอียดของบริการบางครั้งก็ประกาศให้ผู้ขอใช้บริการทราบโดยตรงหรือ ประกาศไว้ในแหล่งค้นหาบริการ (Kregeri, H. 2001:8)

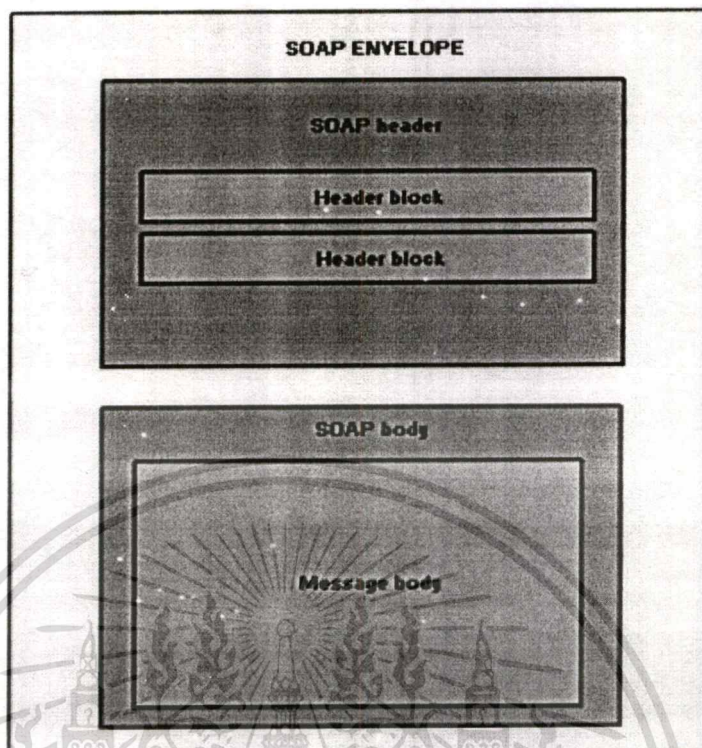
2.1.5 เทคโนโลยีและโปรโตคอลในเว็บเซอร์วิส

เว็บเซอร์วิสจะอยู่บนพื้นฐานของการผสมผสานกันระหว่างมาตรฐานต่างๆที่สามารถทำให้ client สามารถสื่อสารกับผู้ให้บริการและเข้าใจซึ่งกันและกันได้แม้จะอยู่ในหลายแพลตฟอร์มและหลากหลายภาษา ในตอนนี้จะอธิบายถึงโปรโตคอลและเทคโนโลยีที่ทำให้เกิดเว็บเซอร์วิสพอสังเขป

2.1.5.1 SOAP

จากที่ได้อธิบายความหมายและโครงสร้างของ XML ทำให้เห็นว่า XML มีประสิทธิภาพในการแลกเปลี่ยนข้อมูล แต่ก็ยังไม่เพียงพอต่อการแลกเปลี่ยนข้อมูลกันบนเว็บ จึงจำเป็นต้องมีโปรโตคอลสำหรับจัดการรูปแบบของเอกสาร XML และผู้รับเข้าใจว่าจะอะไรเป็นส่วนหลักจากที่ได้รับมา ที่เรียกว่า payload ซึ่งเป็นส่วนหนึ่งของข้อความ ดังนั้นจึงเกิดโปรโตคอลที่เรียกว่า Simple Object Access Protocol (SOAP) เป็นโปรโตคอลที่มีพื้นฐานมาจาก XML เพื่อใช้ในการแลกเปลี่ยนข่าวสารในสภาพแวดล้อมแบบกระจาย SOAP จะเป็นรูปแบบข้อความทั่วไปสำหรับการ

เอกสารในการแลกเปลี่ยนข้อมูลระหว่าง clients กับบริการ โดยที่ข้อความ SOAP จะประกอบไปด้วย
ไม่ว่าองค์ประกอบดังรูปที่ 2.2 คือ SOAP envelop SOAP header และ SOAP body envelope



รูปที่ 2.2 โครงสร้างของ SOAP

SOAP body envelope จะประกอบไปด้วย 2 สิ่งคือ XML namespace และ encoding style XML namespace จะระบุชื่อที่สามารถใช้ในข้อความ XML namespaces ออกแบบมาเพื่อหลีกเลี่ยงการชนกันของชื่อ คือ ชื่อที่เหมือนกันสามารถใช้ต่าง items กันได้ โดยต้องใช้ต่าง namespace กัน encoding style ใช้ระบุชนิดของข้อมูลโดยข้อความ SOAP ถ้ามีการปรากฏ SOAP header ของเอกสาร มันเป็นตัวที่ขยายให้ข้อความ SOAP มีลักษณะเป็นโมดูล และเนื่องจากข้อความ SOAP จะต้องมีการสื่อสารผ่าน node ตัวกลางจนถึง node ปลายทาง ซึ่งในแต่ละ node ตัวกลางจะมีโอเปอเรชันของตัวเองที่เพิ่มเข้ามาจัดการกับข้อความ SOAP เช่น การปรับเปลี่ยนข้อความ โอเปอเรชันเกี่ยวกับความปลอดภัย ดังนั้น SOAP header จะเป็นตัวระบุว่าจะมีโอเปอเรชันอะไรบ้างที่กระทำกับข้อความตรง Node ตัวกลาง ซึ่งอิสระจากโอเปอเรชันตรง node ปลายทาง ในส่วนของ SOAP body จะเป็นส่วนหลักของข้อความ SOAP และเป็นส่วนสุดท้ายในข้อความ SOAP ที่ผู้รับทำไปใช้ประโยชน์

ด้านล่างเป็นตัวอย่างของข้อความ SOAP ออกแบบมาเพื่อใช้ในการค้นหาราคาของหนังสือ สิ่งที่เกิดขึ้นคือ อีลีเมนต์ <SOAP-ENV : Envelope> <SOAP-ENV : Header> และ <SOAP-ENV : Body> เป็นตัวบอกว่า เป็น envelope header และ body ของข้อความ SOAP ตรงค่า attribute ของ mustUnderstand=1 ในส่วนหัวหมายความว่าผู้รับข้อความ SOAP ต้อง

ประมวลผลมัน (Ed Ott, 2005:7)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆก็ตาม หากมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

xmlns:SOAP-ENV=
    "http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:
encodingStyle=
    "http://schemas.xmlsoap.org/soap/encoding/">
    <SOAP-ENV:Header>
        <t:Transaction xmlns:t="some-URI">
            SOAP-ENV:mustUnderstand="1">
        </t:Transaction>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <m:GetBookPrice xmlns:m="some-URI">
            <title>My Life and Times</title>
            <author>Felix Harrison</author>
        </m: GetBookPrice>
    </SOAP-ENV:Body>
</SOAP-Envelope>

```

2.1.5.2 WSDL

WSDL ย่อมาจาก Web Service Description Language (WSDL) ซึ่งเป็นตัวบอกรูปแบบของการร้องขอจะเป็นแบบไหน และสิ่งที่ client ได้รับมีความหมายอย่างไร wsdl เป็นเอกสารที่ใช้อธิบายการติดต่อกับ web service wsdl เป็นตัวกำหนด XML schema สำหรับอธิบาย web service โดยที่ client จะทำการค้นหา wsdl โดยเริ่มจากไปค้นหาที่ UDDI registry จากนั้น UDDI registry จะบอก pointer ของเอกสาร wsdl ที่ต้องการ จากนั้นโปรแกรมเมอร์จะใช้ข้อมูลการติดต่อในเอกสาร wsdl ไปสร้างโปรแกรมเพื่อเรียกใช้บริการ (Ed Ort. 2005 : 8)

2.1.5.3 UDDI

UDDI(Universal Description, Discovery and Integration) หน้าทีให้กับบุคคลสองกลุ่มคือ ผู้ที่ต้องการนำเว็บเซอร์วิส ของตนมาเผยแพร่และเปิดให้บริการ UDDI Registry ต้องอนุญาตให้ผู้ที่มาประกาศ นั้นทำการจัดการเกี่ยวกับเว็บเซอร์วิสที่ตนจะเผยแพร่ไม่ว่าจะเพิ่มเข้าบัญชีรายชื่อ แก้ไข หรือลบออกนอกจากนี้ UDDI Registry จะต้องให้บริการในส่วนของผู้ที่จะมาติดต่อเพื่อค้นหาและขอข้อมูลเกี่ยวกับเว็บเซอร์วิส UDDI Registry ต้องให้บริการในการสืบค้นข้อมูล และให้รายละเอียดของเว็บเซอร์วิสต่างๆ ที่ผู้ที่มาประกาศ ได้นำมาเผยแพร่ไว้ จากนั้นก็จะเป็น

หน้าที่ของ ผู้ร้องขอบริการเองที่ตัดสินใจเลือกใช้เซอร์วิสและทำการติดต่อไปยัง ผู้ที่มาประกาศตามข้อมูลที่ ผู้ที่มาประกาศ ได้เผยแพร่ไว้ (OASIS Open. 2002)

UDDI เป็นมาตรฐานที่มีไว้ใช้สำหรับค้นหาเว็บเซอร์วิส ซึ่งกลุ่ม <http://www.uddi.org> เป็นกลุ่มที่เสนอมาตรฐานนี้ และกำหนดให้ UDDI เป็นมาตรฐานสำหรับการทำงานข้ามแพลตฟอร์มของ B2B (Business to Business Interoperability)

UDDI Registry จะให้ข้อมูลสำหรับผู้ที่ต้องการค้นหาเว็บเซอร์วิส 3 ลักษณะ ซึ่งเปรียบเทียบได้กับสมุดโทรศัพท์ ดังนี้

1. สมุดหน้าขาว (White Page) – เป็นข้อมูลเกี่ยวกับตัวบริษัทและข้อมูลผู้ประกาศ ได้แก่ ชื่อบริษัท, คำอธิบายรายละเอียด ข้อมูลการติดต่อ และหมายเลขอุตสาหกรรม
2. สมุดหน้าเหลือง (Yellow Page) – เป็นข้อมูลเกี่ยวกับการจัดหมวดหมู่ของอุตสาหกรรมและบริการ
3. สมุดหน้าเขียว (Green Page) – เป็นข้อมูลทางด้านเทคนิคต่างๆ

2.2 XML

2.2.1 ความหมายของ XML

eXtensible Markup Language (XML) ได้กลายมาเป็นมาตรฐานสำคัญสำหรับการการอธิบายข้อมูลที่ใช้แลกเปลี่ยนกันบนเว็บ จากชื่อของ XML ข้างต้นทำให้ทราบว่า XML เป็น markup language ประกอบไปด้วย tags ที่ใช้ mark up เนื้อหาในเอกสารเพื่ออธิบายเนื้อหาในเอกสาร XML tags ใช้ระบุสารสนเทศในเอกสาร และใช้ระบุโครงสร้างของสารสนเทศด้วย จากตัวอย่าง XML mark up ด้านล่างใช้ระบุสารสนเทศของ bookshelf XML markup ใช้อธิบายโครงสร้างของ bookshelf ด้วย โครงสร้างของ bookshelf ประกอบไปด้วย 2 items ใช้ระบุ book และ book ก็จะมี 3 items คือ title author และ price

```
<bookshelf>
  <book>
    <title>My Life and Times</title>
    <author>Felix Harrison</author>
    <price>39.95</price>
  </book>
</bookshelf>
```

เอกสารนี้เป็นเอกสารตัวอย่าง แม้ว่า tags เช่น <book> และ <title> จะปรากฏเป็นลำดับชั้นและให้ความหมายแก่เอกสาร คำว่า XML tags จะไม่มีความหมายโดยตัวมันเอง แต่จะมีความหมายเมื่อมนุษย์เป็นคนใช้

กำหนดให้ในแต่ละ tags เช่น มีความเห็นว่า <book> ใช้ระบุว่าเป็นหนังสือ และ<title><author> และ <price> ใช้ระบุว่าเป็นชื่อเรื่อง ผู้แต่ง และราคาของหนังสือตามลำดับ เอกสาร XML จะใช้สื่อสารระหว่างแอปพลิเคชันซึ่งต้องเข้าใจความหมายที่อยู่ใน tags ด้วยกันทั้งคู่ แอปพลิเคชันต้องสามารถแปลความหมายของ tags ได้ เอกสาร XML จะมีโครงสร้างรูปแบบที่ดี (well-formed structure) จากตัวอย่าง ในแต่ละ XML tags จะต้องมีการปิด ตัวอย่างเช่น <bookshelf>...</bookshelf> และทุกๆ tags เกิดขึ้นภายใน tag อื่น ต้องปิดก่อนที่ tag อื่นดังกล่าวจะปิดลง นั่นคือ </title> </author> และ</price>ต้องมาก่อน </book> โดยทั่วไปเอกสาร XML จะทำงานร่วมกับ schema ที่ใช้ในการระบุกฎไวยากรณ์ของ เอกสาร XML นั่นคือ schema ใช้ระบุว่ามี tags อะไรบ้างที่อนุญาตให้มีได้ในเอกสาร XML รวมถึงโครงสร้างของ tags และกฎอื่นๆ เกี่ยวกับ tags ตัวอย่างเช่น ชนิดข้อมูลอะไรที่จะปรากฏใน tag (หรือไม่มีข้อมูลอะไรถ้ามันเป็น tag ว่าง) เอกสาร XML ที่ถูกต้องจะต้องเป็น well-formed และสอดคล้องกับ schema ที่จะใช้ในการประมวลผลเอกสาร XML ด้วยเหตุนี้ XML จึงถูกนำมาใช้เป็นภาษาข้อมูลในเว็บเซอร์วิส (Ed Ort. 2005:7)

2.2.2 XML Schema

XML Schema คือ ภาษาที่ทำหน้าที่ตรวจสอบความถูกต้องให้กับข้อมูลที่จัดเก็บอยู่ในรูปแบบแหล่งข้อมูล XML เช่นเดียวกับภาษา DTD จะใช้ร่วมกับข้อมูลที่ถูกรวบรวมมาจากฐานข้อมูล XML Schema มีลักษณะของตัวภาษาด້ว้ๆ กับภาษา XML และภาษา XSL ทำให้ไม่เกิดความสับสนมากนักในการศึกษาและการเรียกใช้งาน และที่สำคัญที่สุดคือ XML Schema รองรับการใช้เนมสเปซ (namespace) ซึ่งถือเป็นคอนเซ็ปต์ใหม่ และมีประโยชน์อย่างมากในการพัฒนา XML

ใน XML Schema จะแบ่งอิลิเมนต์ออกเป็น 2 แบบ คือแบบ Simple Type และแบบ Complex Type โดยที่ Simple Type คือ อิลิเมนต์ที่มีข้อมูลภายในเป็นข้อมูลพื้นฐาน อย่างเช่น สตริง , ตัวเลข เป็นต้น ส่วน Complex Type คือ อิลิเมนต์ที่มีแอตทริบิวต์หรือมีข้อมูลภายในเป็นอิลิเมนต์อื่นๆ ดังตัวอย่างต่อไปนี้

```
<xsd:complexType name="USAddress">
```

```
<xsd:sequence>
```

```
<xsd:element name="name" type="xsd:string"/>
```

```
<xsd:element name="street" type="xsd:string"/>
```

```
<xsd:element name="city" type="xsd:string"/>
```

```
<xsd:element name="state" type="xsd:string"/>
```

```
<xsd:element name="zip" type="xsd:decimal"/>
```

```
</xsd:sequence>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<xsd:attribute name="country" type="xsd:NMTOKEN" />
```

```
</xsd:complexType>
```

XML Schema ตามตัวอย่างข้างต้นนี้กำหนดอิลิเมนต์ประเภท USAddress ขึ้นมาเป็นแบบ Complex Type ซึ่งประกอบไปด้วยอิลิเมนต์ name, street, city, state, zip และมีแอตทริบิวต์ country ด้วย นอกจากนี้ยังระบุชนิดข้อมูลของแต่ละอิลิเมนต์ ซึ่งส่วนใหญ่เป็นข้อมูลชนิดสตริง ยกเว้น zip ที่เป็นตัวเลขทศนิยม (decimal) และ country เป็นชนิด NMTOKEN ซึ่งก็คือสตริงประเภทที่ไม่มีช่องว่าง และตั้งชื่อตามกฎของ XML (สราวุธ อ้อยศรีสกุล. 2544 : 104-105)

2.2.3 XSL

XSL ย่อมาจาก eXtensible Stylesheet Language เป็นภาษาสำหรับการจัดรูปแบบของข้อมูล XML เพื่อการแสดงผลผ่านเบราว์เซอร์ การทำงานของ XSL มีลักษณะคล้ายกับ CSS (Cascading Style Sheet) ที่ใช้กำกับรูปแบบการแสดงผลของภาษา HTML นั่นเอง

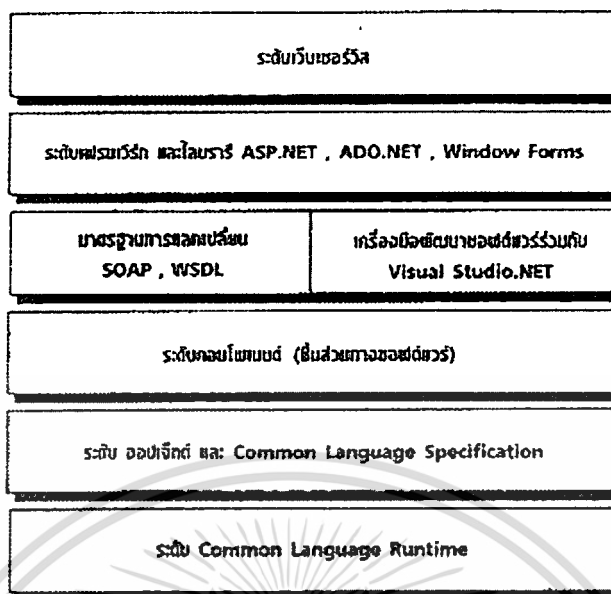
คำว่า XSL เป็นคำเรียกมาตรฐานของการแปลงไฟล์ XML ให้แสดงผลออกทางเบราว์เซอร์ แต่จริงๆ แล้วภาษาที่ใช้สำหรับแปลงไฟล์จะเรียกว่า XSTL ซึ่งจะแปลงไฟล์ XML ซึ่งถือว่าเป็นไฟล์ต้นแบบให้เป็นไฟล์ XHTML ซึ่งสามารถแสดงผลออกเบราว์เซอร์ได้ เราเรียกไฟล์ XHTML ว่าเป็นไฟล์ปลายทาง (นรินทร์ โอพารกิจนันต์. ม.ป.ป. : 157)

2.3 เทคโนโลยี.NET

.NET (อ่านว่า ค็อตเน็ต) เป็นแพลตฟอร์มทางภาษาแบบเปิดสำหรับการพัฒนาประยุกต์ระบบงานในองค์กร และงานพัฒนาเว็บสำหรับการใช้งานสมัยใหม่ ที่เปรียบพร้อมคุณสมบัติและความต้องการอย่างครบถ้วน

จุดประสงค์ของเทคโนโลยี .NET จึงมีไว้สำหรับผู้พัฒนาแบบมืออาชีพที่จะพัฒนาระบบงานสำหรับองค์กรตั้งแต่งานขนาดเล็กจนถึงงานระดับใหญ่มากในองค์กรและระหว่างองค์กร โดยครอบคลุมการประยุกต์งานทั้งแบบดั้งเดิม ที่เรียกว่าแบบโฮสเบส แบบคอนโซลแอปพลิเคชัน แบบไคลเอนต์เซิร์ฟเวอร์ แบบหลายเทียร์ (multi tier) และแบบประยุกต์บนเว็บ หากผู้พัฒนาระบบงานได้ทดลองใช้หรือศึกษาจากเครื่องมือ เช่น การใช้ภาษา C# จะพบว่าเมื่อใช้พัฒนาระบบงานจะมีเครื่องมือช่วยทำให้การพัฒนาได้หลากหลายรูปแบบ ตั้งแต่คอนโซลแอปพลิเคชัน วินโดวส์แอปพลิเคชัน หรือ เว็บแอปพลิเคชัน

ไมโครซอฟท์ได้ทางสถาปัตยกรรมของเทคโนโลยี .NET ไว้เป็นกระดัดดังรูปที่ 2.3



รูปที่ 2.3 โมเดลของเทคโนโลยี .NET

ระดับเว็บเซิร์ฟเวอร์ เป็นระดับบนสุดของ .NET ที่ใช้สำหรับการให้บริการกับผู้ใช้ ระดับนี้เป็นระดับการทำงานโดยผู้ใช้ติดต่อผ่านเว็บ เพื่องานต่างๆ การบริการบนเว็บเน้นให้มีการโต้ตอบผ่านเว็บแบบสองทิศทาง ทั้งที่ติดต่อโดยผู้ใช้ หรือติดต่อระหว่างคอมพิวเตอร์กับคอมพิวเตอร์เองโดยอัตโนมัติ

ระดับเฟรมเวิร์กและไลบรารี เป็นกลุ่มของเฟรมเวิร์กและโปรแกรมที่เก็บเป็นไลบรารี ที่ให้ผู้พัฒนาเรียกไปใช้ กลุ่มนี้ทำให้การพัฒนาระบบงานกระทำได้ง่าย ดี และรวดเร็ว ตัวเฟรมเวิร์กประกอบด้วย ASP.NET (Active Server pages) ADO.NET (ActiveX Data Objects และยังรวมถึงไลบรารีที่ทำให้ผู้พัฒนาระบบทำงานได้ดี โดยเฉพาะในเรื่องวินโดวส์ฟอร์ม เป็นการเขียนโปรแกรมผ่านฟอร์มของวินโดวส์ที่มีกราฟิกส์ต่างๆ ให้เราหยิบมาใช้ได้

มาตรฐานการแลกเปลี่ยนข้อมูล เป็นการสร้างมาตรฐานการแลกเปลี่ยนข้อมูลบนฐาน XML ที่ทำให้ไม่ขึ้นกับแพลตฟอร์มที่ใช้ ส่วนสำคัญในส่วนนี้ประกอบด้วย SOAP-Simple Object Access Protocol เป็นเสมือนส่วนที่อยู่ตรงกลางสำหรับการเชื่อมโยงกัน โดยมีการนิยามสำหรับการเชื่อมโยงกับเว็บ เพื่อให้เบรเซอร์ทำงานได้ทันที ส่วนนี้เราเรียกว่า WSDL - Web Services Description Language

เครื่องมือสำหรับการพัฒนาระบบงาน มีการสร้างสิ่งแวดล้อมให้ผู้พัฒนาระบบงานทำได้ง่าย ส่วนนี้ไมโครซอฟท์ได้สร้าง Visual Studio ขึ้นมาใหม่ และให้ชื่อว่า Visual Studio.NET เป็นเครื่องมือที่ใช้สร้างซอฟต์แวร์ร่วม โดยมีคอมไพเลอร์ที่จะสร้างออปเจ็กต์ต่าง ๆ ภายใต้เครื่องมือนี้ ทำให้เราสร้างระบบงานบนวินโดวส์ (Window Form Application) การสร้างระบบงานบนเว็บ (Web Application) การสร้างระบบงานบนคอนโซล (Console Application) การสร้างเครื่องมือ

เหล่านี้ไมโครซอฟท์ได้สร้างฐานร่วมกัน โดยผู้พัฒนาจะใช้ภาษา () ได้หลายภาษา เช่น C# C++, VB

โมเดลคอมโพเนนต์ ก่อนการพัฒนา .NET มีการวางโครงสร้างโมเดลคอมโพเนนต์กันแล้ว โดยจากกลุ่มใหญ่สามกลุ่ม ซึ่งได้แก่ Cobra จากกลุ่ม Object Management Group อีกมาตรฐานหนึ่งคือ J2EE ของบริษัทซันไมโครซิสเต็ม และกลุ่มสุดท้ายคือไมโครซอฟท์ได้วางมาตรฐานคอมโพเนนต์ที่ชื่อว่า COM สำหรับ .NET ได้นำเอาลักษณะจากหลายกลุ่มมารวมกัน โดยใช้แนวคิดของออปเจ็กต์โอเรียนเตด โครงสร้างของ .NET ทำให้เราสามารถสร้างคอมโพเนนต์ที่ประกอบด้วยคลาสจำนวนมาก ที่มีการกำหนดการเชื่อมต่อหรืออินเตอร์เฟซไว้ชัดเจน ซึ่งโมเดลของ .NET นี้จึงแตกต่างจากของ COM ที่ไมโครซอฟท์เคยกำหนดมาก่อน

โมเดลออปเจ็กต์ เป็นระดับที่สร้างอยู่บนมโนภาพที่ให้ทุกสิ่งทุกอย่างเป็นออปเจ็กต์ภายใต้แนวคิดของออปเจ็กต์โอเรียนเตด เพื่อที่จะกำหนด สเปกของภาษาร่วมกันได้ เพื่อให้ออปเจ็กต์ส่วนนี้เป็นกลาง ไม่ว่าจะมาจากภาษาใด ก็จะได้ตามกรอบของสเปกที่กำหนดไว้ นับเป็นโมเดลที่ทำให้มีการทำงานร่วมกัน และโอนย้ายเพื่อการคำนวณบนเครือข่ายได้ดี

ระดับ Common Language Runtime เป็นระดับที่จะทำการประมวลผลให้เกิดการทำงาน ความจริงระดับนี้เสมือนเป็นเครื่องจักรที่ทำงานตามที่ต้องการโดยมีการสร้างรหัสคำสั่งเสมือนเป็นภาษาเครื่องให้ทำงานได้ จึงทำให้ .NET เป็นกลไกที่จะทำงานได้ตั้งแต่เครื่อง PDA ปาล์ม พ็อกเก็ตพีซี โทรศัพท์มือถือ พีซี หรือเครื่องอื่น ๆ โดยไม่ขึ้นกับฮาร์ดแวร์และโอเอส แนวคิดนี้ก็เหมือนกับจากาที่มี JVM ทำงานได้กับทุกเครื่องจักรนั่นเอง (ยีน ภู่วรรณ. 2550)

2.4 Apache AXIS

AXIS เป็นเครื่องมือที่ใช้ในการพัฒนาเว็บเซอร์วิสทางฝั่งเซิร์ฟเวอร์ และไคลเอนต์ตัวหนึ่ง ซึ่งจะช่วยจัดการกับ SOAP และ WSDL ทำให้ผู้ใช้ AXIS ในการพัฒนาไม่จำเป็นต้องพัฒนา SOAP Request, SOAP Response และ WSDL เอง โดยที่ AXIS จะช่วยจัดการในส่วนนั้นให้ โดยอัตโนมัติ นั่นคือ เราสามารถพัฒนาเว็บเซอร์วิส หรือแอปพลิเคชันเพื่อเรียกใช้เว็บเซอร์วิส โดยการเรียกใช้งานผ่าน API ของ AXIS

2.4.1 โครงสร้างและคอมโพเนนต์ของ Axis

โดยทั่วไป โครงสร้างของ Axis ประกอบไปด้วยคอมโพเนนต์ต่างๆ คือ ระบบย่อยที่ทำงานร่วมกันเพื่อเป็น Server หรือ Client ขึ้นอยู่กับว่าสภาพแวดล้อมของเว็บเซอร์วิส เป็นผู้ให้บริการหรือผู้ร้องขอบริการ (Nagappan, R. et al. 2003 : 149)

2.4.2 จักรกล Axis

จักรกล Axis จะเป็นสภาพแวดล้อมขณะทำงานของ SOAP เพื่อประมวลผลข้อความที่เข้า

มาหรือออกไป โดยดูจากกิจกรรมตรง header ของ SOAP ในการขนส่งในการประมวลผลข้อความ จักรกล Axis ใช้ประโยชน์จากชุดของ handler ที่ต่อกันเป็นโซ่ในการร้องขอและ

ประมวลผลข้อความ ข้อความที่ถูกส่งผ่านไปยัง handler เพื่อการร้องขอคือ อ็อบเจ็กต์ MessageContext (Nagappan, R. et al. 2003 : 150)

2.4.3 Handlers และ Chains

จักรกล Axis จะเป็นตัวประมวลผลข้อความทั้งฝั่ง client และ server ด้วย handler ฝั่ง client และ handler ฝั่ง server จักรกล Axis จะประมวลผลข้อความโดยใช้ชุดของ handler ร้องขอ และหลังจากร้องขอบริการเป้าหมายเสร็จแล้ว ก็จะส่งกลับผ่านชุดของ handler ตอบสนอง Axis จะประกอบด้วยชุดของ handlers ที่มีหน้าที่รับผิดชอบการทำงานที่เหมือนประกอบเข้าด้วยกันเป็นโซ่ Axis จะใช้ handlers บริการในการใช้ประโยชน์ RPC และเว็บเซอร์วิสแบบข้อความ (messaging-based web services)

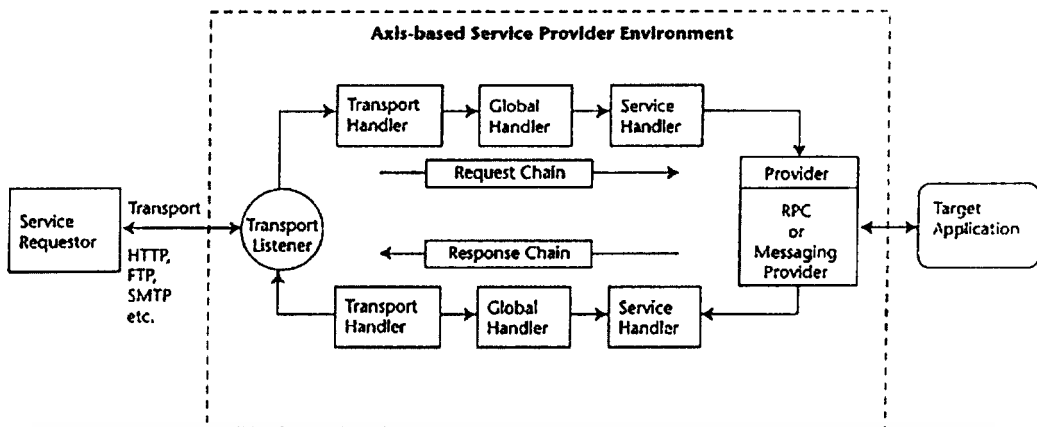
คุณลักษณะหลักของ handlers บริการของ Axis สำหรับ RPC และเว็บเซอร์วิสแบบข้อความ เป็นดังต่อไปนี้

ในลักษณะการทำงานในรูปแบบประเภท RPC ของเว็บเซอร์วิส handler บริการ org.apache.axis.provider.java.RPC.Provider จะกำหนดเมทอดที่จำเป็นสำหรับการร้องขอ และประมวลผลโดยให้พารามิเตอร์บรรจุเป็นส่วนหนึ่งของข้อความร้องขอ SOAP มันจะใช้ serialization และ deserialization ของ JAVA อ็อบเจ็กต์ ไปเป็น XML และจาก XML ไปเป็น java อ็อบเจ็กต์ ระหว่างการร้องขอและการตอบสนองบริการ

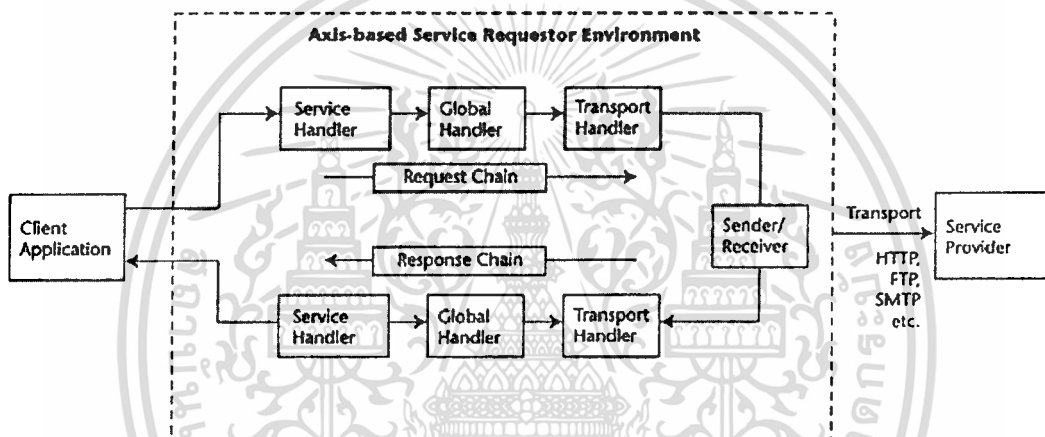
ในลักษณะประเภทข้อความ (รูปแบบเอกสาร) ของบริการ handler บริการ org.apache.axis.providers.java.MsgProvider จะเรียกเมทอดและส่งผ่านเอกสาร XML ที่บรรจุในข้อความ SOAP

จุดน่าสังเกตในบริการรูปแบบ RPC handler บริการทำการประมวลผลข้อความโดยใช้ serialization และ deserialization ในการเปลี่ยน XML เป็น JAVA อ็อบเจ็กต์ และจาก JAVA อ็อบเจ็กต์เป็นเอกสาร XML ด้วยบริการในรูปแบบเอกสาร ข้อมูล XML จะส่งผ่านไปยังเมทอดเป้าหมายเพื่อประมวลผล ระหว่างการประมวลผล handlers บริการปรากฏ SOAPFaults ด้วย เป็น AxisFault exceptions (Nagappan, R. et al. 2003 : 150)

รูปที่ 2.4 แสดงโครงสร้างทางด้าน server ของจักรกล Axis ด้วย handlers และโซ่ และรูปที่ 2.5 แสดงโครงสร้างทางด้าน client ของจักรกล Axis ด้วย handlers และโซ่



รูปที่ 2.4 โครงสร้างพื้นฐานของผู้ให้บริการบนพื้นฐานของ Axis



รูปที่ 2.5 โครงสร้างพื้นฐานของผู้ร้องขอบริการบนพื้นฐานของ Axis

2.4.4 การบริหาร Axis

การบริหาร Axis จะสามารถบริหารและกำหนดค่าข้อมูลสำหรับจักรกล Axis ในการทำให้โซ่บริการและบริการ SOAP อื่นๆทำงาน มันอนุญาตให้สภาพแวดล้อมของจักรกล Axis สามารถถูกกำหนดค่าได้ด้วยไฟล์ Web Service Deployment Descriptor (WSDD) ไฟล์ WSDD จะกำหนดการขนส่งที่สนับสนุน การกำหนด global ของจักรกล Axis เป็น handlers และการ deploy บริการ และส่วนมากมีรูปแบบเป็น server-codfig.wsdd ซึ่งประกอบไปด้วยข้อมูลเกี่ยวกับการติดตั้งจักรกล Axis และการขนส่งที่สนับสนุน บริการ handlers เป็นต้น เมื่อเรารันคำสั่งต่อไปนี้

Java.org.apache.axis.client.Administrator list

มันจะให้รายการที่รวมไปถึง บริการการบริหาร Axis ที่ถูก deploy เป็น AdminSer Axis จะอนุญาตให้บริหารทางไกลได้ (remote) ด้วยการกำหนดค่าโดยกำหนดให้พารามิเตอร์ enableRemoteAdmin เป็น true (Nagappan, R. et al. 2003 : 151-152)

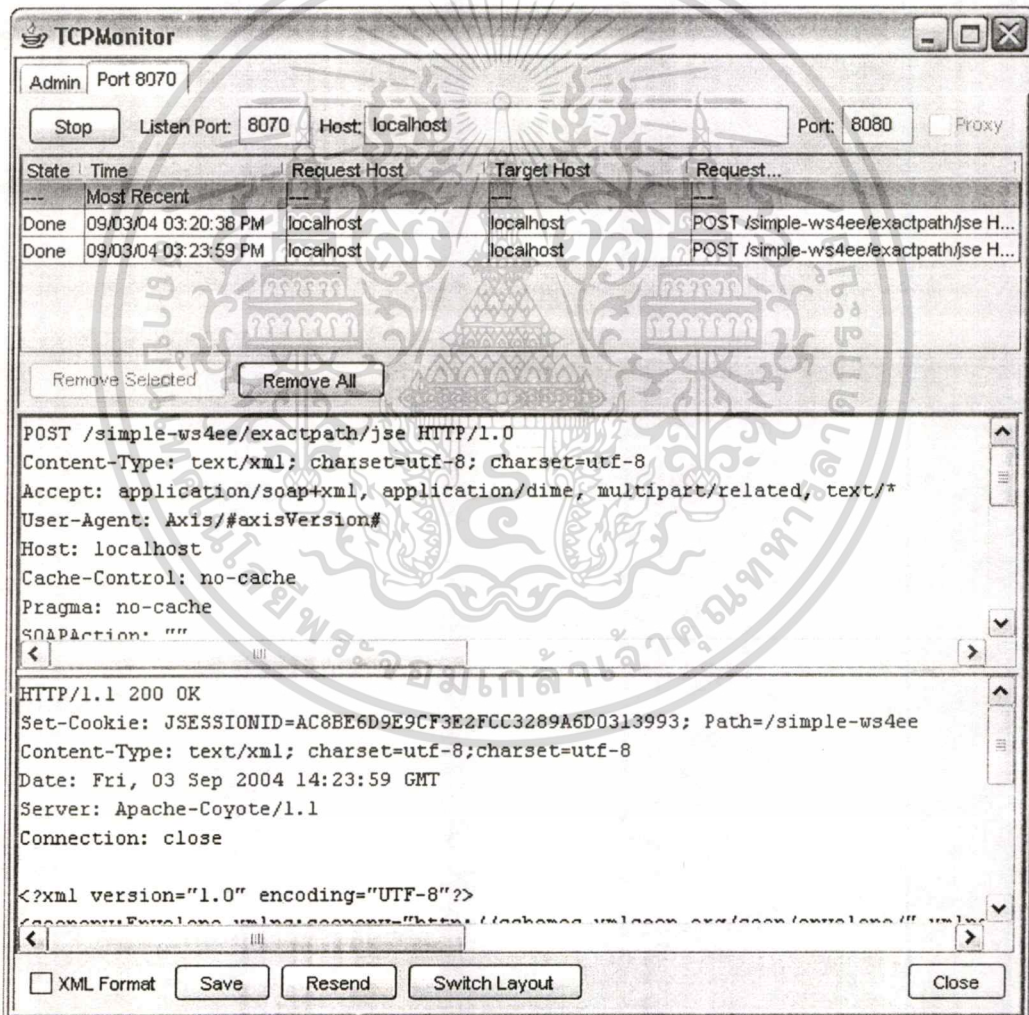
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 Axis TCP Monitor

Axis จะให้ tcpmon ซึ่งเป็น TCP monitor มีประโยชน์เพื่อให้สามารถดู บันทึก และการแก้ไขของ SOAP ทั้งการร้องขอและตอบสนอง มันช่วยให้สภาพแวดล้อมของเว็บเซอร์วิสที่พัฒนาด้วย Axis สามารถตรวจจับ SOAP ทั้งการร้องขอและตอบสนองในเวลาจริง และอนุญาตให้การทดสอบของ SOAP การร้องขอโดยตัดแต่งและส่งทำงานต่อไปได้ด้วย (Nagappan, R. et al. 2003 : 153)

การใช้ tcpmon จำเป็นต้องอาศัย port ของ client โฮสต์ SOAP เป้าหมาย และ port ของมัน tcpmon จะแสดงการสัญจรของ SOAP การร้องขอและการตอบสนองและแสดงในรูปของ GUI

สำหรับรูปแบบ GUI ของ tcpmon แสดงดังรูปที่ 2.6



รูปที่ 2.6 GUI ของ tcpmon

การร้องขอสามารถแก้ไขและทำการส่งใหม่ได้ ซึ่งจะทำให้สามารถศึกษาพฤติกรรมและดู

ข้อความที่ตอบกลับมาโดยไม่ต้องเปลี่ยนแอปพลิเคชันของ client

เอกสารเป็นเอกสารทบทวนวิชาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

tcpmon สามารถบันทึกการร้องขอและตอบสนองใน XML ไปเป็น text file ได้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Google Maps API

Google Maps API ช่วยให้เราสามารถพัฒนาโปรแกรมเพื่อแทรก Google Maps เข้าไปเป็นองค์ประกอบส่วนหนึ่งในเว็บเพจที่ต้องการได้โดยเขียนเป็นรหัส html และ JavaScript ในรูปแบบที่ไม่สลับซับซ้อนนักสำหรับงานแผนที่ง่ายๆ Google Maps API มีขีดความสามารถกว้างขวางเน้นในด้านการนำเสนอข้อมูลแผนที่ในลักษณะ Push pin หรือ Place marker ซึ่งสามารถกำหนดให้แสดงข้อมูลประกอบแผนที่เมื่อผู้คลิกที่ตัว pushpin /marker นั้นๆ บริการเรื่องแผนที่ของ Google นี้เริ่มต้นตั้งแต่กลางปีค.ศ. 2005 เป็นบริการฟรี จัดให้แก่ผู้ใช้ที่ร้องขอโดยคาดหวังที่จะใช้การโฆษณาบนแผนที่เป็นรายได้กลับคืน แต่ในระยะแรกจะยังไม่มีโฆษณาดังกล่าว (สวัสดีชัย เกรียงไกรเพชร. 2549)

เนื่องจากจัดทำ Google Maps API เป็นโปรแกรมรหัสเปิด (Open source program) ในภาษาจาวาสคริปต์ จึงทำให้ผู้ใช้ที่เป็นนักพัฒนาโปรแกรมสามารถเข้าไปดูรายละเอียดของรหัสโปรแกรมได้สะดวก รวมทั้งสามารถปรับเปลี่ยนแก้ไขโปรแกรมได้ ทำให้ Google Maps API มีผู้ใช้งานอย่างกว้างขวาง เหตุผลสำคัญอีก 2 อย่างที่ส่งเสริมให้มีผู้ใช้มากคือแผนที่และภาพถ่ายดาวเทียมคุณภาพดีที่ใช้สนับสนุนการทำแผนที่ที่มีให้ครอบคลุมพื้นที่ต่างๆ อย่างกว้างขวาง และชื่อเสียงของโปรแกรม Google Earth เสริมด้วยบริการ Google Local ที่มีมาก่อน ลักษณะของ Google Maps API ที่ปรากฏในเว็บไซด์เป็นดังรูปที่ 2.7



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์โดย Google Inc. การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจาก Google Inc. อาจเป็นความผิดตามกฎหมาย

รูปที่ 2.7 ตัวอย่างเว็บแอปพลิเคชันที่สร้างด้วย Google Map API

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์และออกแบบระบบ

Diagram illustrating the architecture of a web service system for a Thai government website. The system consists of the following components and interactions:

- User**: Interacts with the **User Interface**.
- User Interface**: Connects to the **Central Web Service**.
- Central Web Service**: The core service that interacts with the **Database** and the **Web Service**.
- Database**: Connected to the **Central Web Service**.
- Web Service**: Connected to the **Central Web Service**.
- XML with SOAP Protocol**: A protocol used for communication between the **Central Web Service** and external services.
- Web service รายที่ 1**: An external web service connected via XML with SOAP Protocol.
- Database รายที่ 1**: A database connected to **Web service รายที่ 1**.
- Google map APIs**: A service used by the **Central Web Service**.
- Web Service รายที่ 2**: An external web service connected via XML with SOAP Protocol.
- Database รายที่ 2**: A database connected to **Web Service รายที่ 2**.

รูปที่ 3.1 ภาพรวมของระบบเว็บเซอร์วิสเพื่อการส่งของอสังหาริมทรัพย์

โครงการนี้จะมีการออกแบบและพัฒนาระบบด้วยกัน 3 ระบบ ดังนี้

1. ระบบเว็บเซอร์วิสของเจ้าของโครงการรายที่ 1 ประกอบด้วยการร้องขอข้อมูลโครงการ การส่งจองข้อมูลโครงการ การตรวจสอบสถานะการแจ้งจองโครงการ โดยจะพัฒนาด้วยเทคโนโลยี Java API for XML-based RPC (JAX-RPC) และใช้ฐานข้อมูล MySQL
2. ระบบเว็บเซอร์วิสของเจ้าของโครงการรายที่ 2 ประกอบด้วยการร้องขอข้อมูลโครงการ การส่งจองข้อมูลโครงการ การตรวจสอบสถานะการแจ้งจองโครงการ โดยจะพัฒนาด้วยเทคโนโลยี Java API for XML-based RPC (JAX-RPC) และใช้

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อวัตถุประสงค์ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

3. ระบบเว็บแอปพลิเคชันที่ทำหน้าที่เป็นเว็บท่า ที่จะใช้ในการเรียกใช้งานเว็บเซอร์วิสของเจ้าของโครงการ ซึ่งจะพัฒนาด้วยเทคโนโลยี .Net โดยเลือก ASP.NET ด้วยภาษา C# เป็นภาษาในการพัฒนา และเลือกใช้ฐานข้อมูล SQLServer 2000

3.1 การออกแบบเว็บท่าเพื่อการเลือกซื้ออสังหาริมทรัพย์

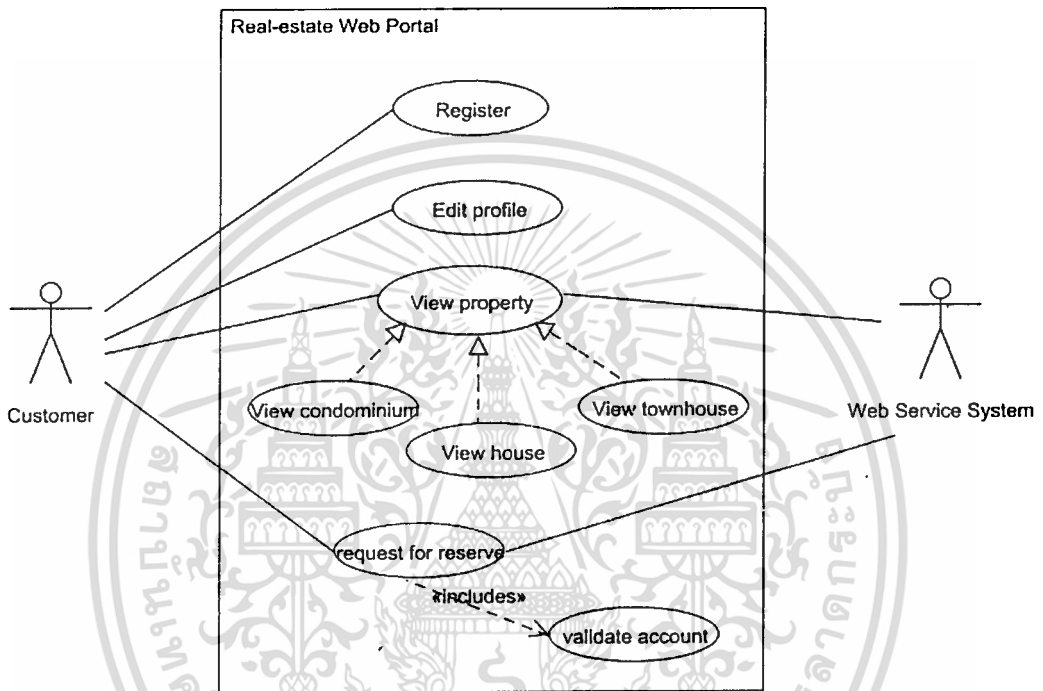
เว็บท่า(web portal) เพื่อธุรกิจอสังหาริมทรัพย์ เป็นเว็บแอปพลิเคชัน ที่ทำหน้าที่เรียกใช้งานเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ต่างๆ โดยใช้ เทคโนโลยี .NET และ ASP.NET ซึ่งติดตั้งอยู่ในระบบปฏิบัติการ Windows XP และใช้ระบบฐานข้อมูล SQLServer 2000

ขั้นตอนการทำงานของ เว็บกลางมีดังนี้

1. เว็บท่าทำการค้นหาข้อมูลการให้บริการของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ทางลงทะเบียนไว้กับ Registry Server
2. เมื่อพบรายชื่อและข้อมูลต่างๆเว็บเซอร์วิสธุรกิจอสังหาริมทรัพย์ทั้ง 2 ราย คือ เว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 2 แล้วทำการบันทึกข้อมูลลงบนฐานข้อมูลของเว็บกลาง
3. เว็บท่าทำการติดต่อกับกับเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่1 และ2
4. เมื่อลูกค้าทำการร้องขอข้อมูล โครงการ ต้องการส่งจองโครงการ หรือตรวจสอบสถานะของการส่งจองโครงการ เว็บกลางจะทำการติดต่อไปยังเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ เพื่อทำการตอบรับการร้องขอตามที่ลูกค้าต้องการ
5. ลูกค้าจะสามารถทราบรายการ โครงการ พร้อมราคาเริ่มต้น แบบบ้าน จากเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ต่างๆ แล้วสามารถเปรียบเทียบราคาจากเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ต่างๆ ได้
6. เมื่อลูกค้าทำการส่งจอง โครงการ เว็บท่าจะทำการบันทึกข้อมูลการสั่งซื้อไว้ภายในฐานข้อมูล พร้อมกับส่งรายการจองไปยังเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายนั้น
7. เมื่อเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ตอบรับการสั่งซื้อกลับมา เว็บกลางจะรับการยืนยันรายการสั่งซื้อนั้น แล้วแจ้งให้ลูกค้าทราบ

3.1.1 ยูสเคสไดอะแกรม

การวิเคราะห์และออกแบบระบบ เริ่มจากรวบรวมความต้องการ ซึ่งได้ 8 ยูสเคส คือ register , edit profile , view property , view condominium , view house , view townhouse request for reserve และ validate account และมี actor 2 actor คือ Customer และ Web service system ดังรูปที่ 3.2



รูปที่ 3.2 ยูสเคสไดอะแกรมของเว็บท่า

3.1.2 Use Case Description

ตารางที่ 3.1 รายละเอียดของยูสเคส Register

Use Case	Register
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ในการลงทะเบียนกับระบบ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าสามารถลงทะเบียนกับระบบได้
เมื่อทำงานไม่สำเร็จ	ลูกค้าไม่สามารถลงทะเบียนกับระบบได้ โดยจะมีการแจ้งเหตุผลที่ไม่สามารถลงทะเบียนได้กลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการลงทะเบียนเข้ามา
อินพุต	ข้อมูลในการลงทะเบียนของลูกค้า
เอาต์พุต	ผลลัพธ์ในการลงทะเบียนของลูกค้า
รายละเอียด	1. ลูกค้าส่งข้อมูลเพื่อขอลงทะเบียนให้กับระบบ 2. ระบบทำการตรวจสอบความถูกต้องของข้อมูล 3. ระบบบันทึกข้อมูลลูกค้าลงฐานข้อมูล 4. ส่งผลลัพธ์กลับไปให้ลูกค้า

ตารางที่ 3.2 รายละเอียดของยูสเคส Edit profile

Use Case	Edit profile
วัตถุประสงค์	เป็นบริการที่ให้ลูกค้าใช้ในการแก้ไขข้อมูลของตัวเอง
เงื่อนไขเมื่อเริ่มต้น	ลูกค้าต้องทำการล็อกอินเข้าสู่ระบบ
เมื่องานทำสำเร็จ	ลูกค้าได้แก้ไขข้อมูลของตัวเองเรียบร้อยแล้ว
เมื่อทำงานไม่สำเร็จ	ข้อมูลของลูกค้าไม่ได้รับการแก้ไข
Actor ที่เกี่ยวข้อง	Customer
สิ่งกระตุ้นการทำงาน	ลูกค้าเลือกเมนูในการแก้ไขข้อมูล
อินพุต	Username ของลูกค้าที่ล็อกอินเข้าสู่ระบบ
เอาต์พุต	ข้อมูลของลูกค้า

ตารางที่ 3.2 รายละเอียดของยูสเคส Edit profile (ต่อ)

รายละเอียด	1. ลูกค้าเลือกเมนูในการแก้ไขข้อมูล 2. ระบบทำการดึงข้อมูลลูกค้ามาแสดง 3. ลูกค้าแก้ไขข้อมูลตัวเอง 4. ระบบทำการถามความแน่ใจในข้อมูลที่ลูกค้าระบุมีความถูกต้องหรือไม่ 5. ระบบทำการบันทึกรายละเอียดของลูกค้า
------------	---

ตารางที่ 3.3 รายละเอียดของยูสเคส View property

Use Case	View property
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ในการร้องขอข้อมูลอสังหาริมทรัพย์กับระบบ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าได้รับรายละเอียดข้อมูลของอสังหาริมทรัพย์แต่ละประเภท
เมื่อทำงานไม่สำเร็จ	ลูกค้าจะไม่ได้รับรายละเอียดข้อมูลของอสังหาริมทรัพย์ โดยจะมีเหตุผลที่ไม่สามารถร้องขอข้อมูลอสังหาริมทรัพย์ ได้กลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการร้องขอข้อมูลอสังหาริมทรัพย์
อินพุต	ข้อมูลเงื่อนไขที่ใช้ในการค้นหาอสังหาริมทรัพย์
เอาต์พุต	ข้อมูลของอสังหาริมทรัพย์ ที่มีคุณสมบัติตรงตามเงื่อนไข
รายละเอียด	1. ลูกค้าส่งรายละเอียดของอสังหาริมทรัพย์ ที่ต้องการค้นหา 2. ระบบทำการร้องขอข้อมูลจากเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ 3. ระบบแจ้งข้อมูลที่ได้รับจากเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ แก่ลูกค้า

ตารางที่ 3.4 รายละเอียดของยูสเคส View condominium

Use Case	View condominium
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ในการร้องขอข้อมูลคอนโดมิเนียมกับระบบ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าได้รับรายละเอียดข้อมูลของคอนโดมิเนียม
เมื่อทำงานไม่สำเร็จ	ลูกค้าจะไม่ได้รับรายละเอียดข้อมูลของคอนโดมิเนียม โดยจะมีเหตุผลที่ไม่สามารถร้องขอข้อมูลคอนโดมิเนียมได้กลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการร้องขอข้อมูลคอนโดมิเนียม
อินพุต	ข้อมูลเงื่อนไขที่ใช้ในการค้นหาคอนโดมิเนียม
เอาต์พุต	ข้อมูลของคอนโดมิเนียมที่มีคุณสมบัติตรงตามเงื่อนไข
รายละเอียด	1. ลูกค้าส่งรายละเอียดของคอนโดมิเนียมที่ต้องการค้นหา 2. ระบบทำการร้องขอข้อมูลจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ 3. ระบบแจ้งข้อมูลที่ได้รับจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ แก่ลูกค้า

ตารางที่ 3.5 รายละเอียดของยูสเคส View house

Use Case	View house
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ในการร้องขอข้อมูลบ้านกับระบบ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าได้รับรายละเอียดข้อมูลของบ้าน
เมื่อทำงานไม่สำเร็จ	ลูกค้าจะไม่ได้รับรายละเอียดข้อมูลของบ้าน โดยจะมีเหตุผลที่ไม่สามารถร้องขอข้อมูลบ้านได้กลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer และ Web Service System
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการร้องขอข้อมูลบ้าน
อินพุต	ข้อมูลเงื่อนไขที่ใช้ในการค้นหามาน
เอาต์พุต	ข้อมูลของคอนโดมิเนียมที่มีคุณสมบัติตรงตามเงื่อนไข

ตารางที่ 3.5 รายละเอียดของยูสเคส View house(ต่อ)

รายละเอียด	1. ลูกค้าส่งรายละเอียดของบ้านที่ต้องการค้นหา 2. ระบบทำการร้องขอข้อมูลจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ 3. ระบบแจ้งข้อมูลที่ได้รับจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ แก่ลูกค้า
------------	---

ตารางที่ 3.6 รายละเอียดของยูสเคส View townhouse

Use Case	View townhouse
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ในการร้องขอข้อมูล ทาวเฮาส์กับระบบ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าได้รับรายละเอียดข้อมูลของทาวเฮาส์
เมื่อทำงานไม่สำเร็จ	ลูกค้าจะไม่ได้รับรายละเอียดข้อมูลของทาวเฮาส์ โดยจะมี เหตุผลที่ไม่สามารถร้องขอข้อมูลทาวเฮาส์ได้กลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer และ Web Service System
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการร้องขอข้อมูลทาวเฮาส์
อินพุต	ข้อมูลเงื่อนไขที่ใช้ในการค้นหาทาวเฮาส์
เอาต์พุต	ข้อมูลของทาวเฮาส์ที่มีคุณสมบัติตรงตามเงื่อนไข
รายละเอียด	1. ลูกค้าส่งรายละเอียดของทาวเฮาส์ที่ต้องการค้นหา 2. ระบบทำการร้องขอข้อมูลจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ 3. ระบบแจ้งข้อมูลที่ได้รับจากเว็บเซอร์วิสของบริษัท อสังหาริมทรัพย์ แก่ลูกค้า

ตารางที่ 3.7 รายละเอียดของยูสเคส Request for reserve

Use Case	Request for reserve
วัตถุประสงค์	เป็นบริการเว็บเซอร์วิสเพื่อให้ลูกค้าใช้ร้องขอเพื่อแจ้งความ สนใจสั่งจองโครงการ
เงื่อนไขเมื่อเริ่มต้น	ลูกค้าต้องทำการล็อกอินเข้าสู่ระบบ
เมื่องานทำสำเร็จ	ระบบทำการบันทึกข้อมูลการร้องขอของลูกค้า ไปยังบริษัท เจ้าของโครงการ และบันทึกเก็บไว้ที่เว็บส่วนกลาง

ตารางที่ 3.7 รายละเอียดของยูสเคส Request for reserve (ต่อ)

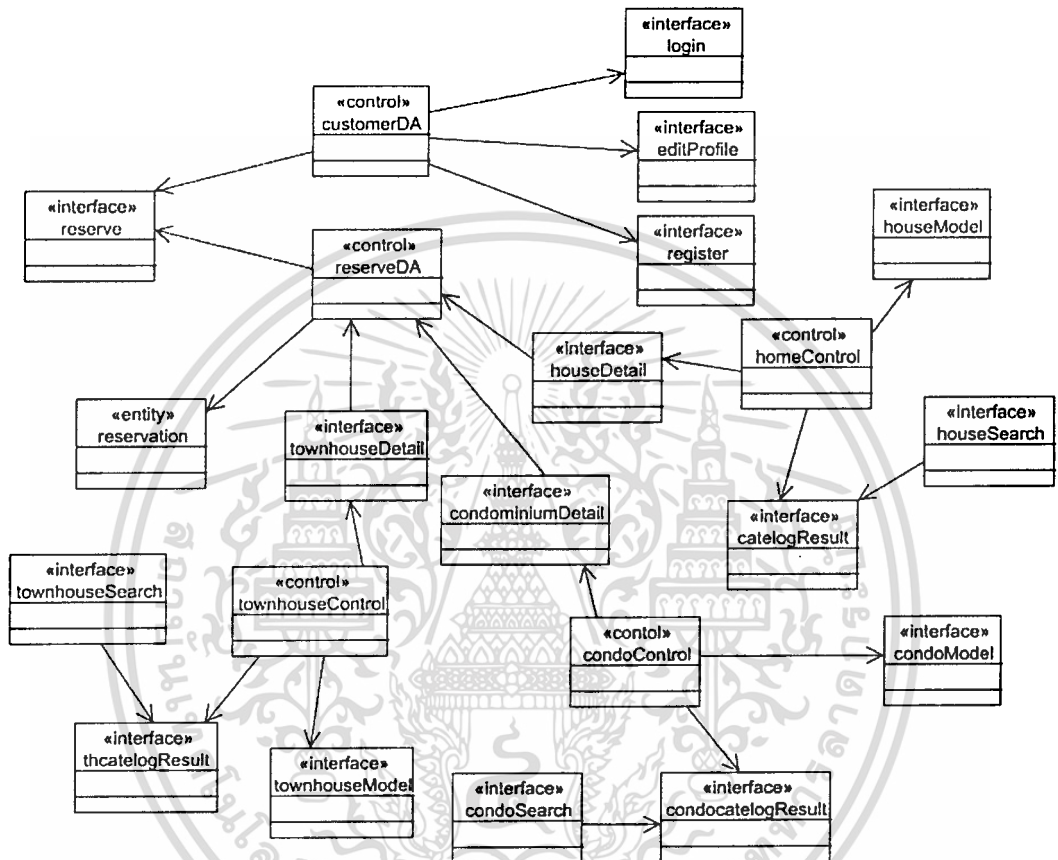
เมื่อทำงานไม่สำเร็จ	ระบบจะไม่ทำการบันทึกข้อมูลการร้องขอของลูกค้า โดยจะมีการแจ้งเหตุผลที่ไม่สามารถส่งจองโครงการกลับไปหาลูกค้า
Actor ที่เกี่ยวข้อง	Customer และ Web Service System
สิ่งกระตุ้นการทำงาน	ลูกค้าเรียกเมธอดในการร้องขอเพื่อแจ้งความสนใจโครงการ
อินพุต	ข้อมูลลูกค้าและโครงการ
เอาต์พุต	ข้อมูลรายละเอียดการแจ้งความจำนงโครงการ
รายละเอียด	1. ลูกค้าส่งข้อมูลการร้องขอโครงการให้แก่ระบบ 2. ระบบทำการร้องขอเมธอดเพื่อแจ้งความสนใจโครงการ จากเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ 3. เว็บกลางทำการบันทึกข้อมูลการจองไว้เป็นหลักฐาน

ตารางที่ 3.8 รายละเอียดของยูสเคส validate account

Use Case	validate account
วัตถุประสงค์	เป็นบริการที่ใช้ในการตรวจสอบรหัสผ่านในการเข้าใช้บริการ
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ลูกค้าสามารถล็อกอินเข้าสู่ระบบได้
เมื่อทำงานไม่สำเร็จ	ลูกค้าไม่สามารถล็อกอินเข้าสู่ระบบได้ โดยจะมีการแจ้งเหตุผลที่ไม่สามารถล็อกอินเข้าสู่ระบบได้
Actor ที่เกี่ยวข้อง	Customer
สิ่งกระตุ้นการทำงาน	ลูกค้าเลือกเมนูล็อกอินเข้าสู่ระบบ
อินพุต	ชื่อผู้ใช้และรหัสผ่านของลูกค้า
เอาต์พุต	แสดงหน้า Welcome ต่อลูกค้า
รายละเอียด	1. ลูกค้าเลือกเมนูล็อกอินเข้าสู่ระบบ พร้อมทั้งใส่ชื่อผู้ใช้และรหัสผ่าน 2. ระบบทำการตรวจสอบความถูกต้องของข้อมูลที่ได้รับ และแสดงหน้าผลลัพธ์การล็อกอินกลับไป

3.1.4 คลาสไดอะแกรม

สำหรับรายละเอียดของคลาสภายในโปรแกรมที่ต้องการสร้างขึ้น และความสัมพันธ์ต่างๆ ระหว่างคลาส โดยคลาสใช้สัญลักษณ์เป็นสี่เหลี่ยมผืนผ้า ภายในแบ่งเป็น ชื่อคลาส คุณสมบัติ และ เมธอดเรียงจากบนลงล่าง ดังรูปที่ 3.3



รูปที่ 3.3 คลาสไดอะแกรมของเว็บกลาง

คลาสภายในโปรแกรมที่ต้องสร้างขึ้น ได้แก่

1. customerDA เป็นคลาสที่ใช้ควบคุมเพื่อการเข้าถึงข้อมูลลูกค้าในฐานะข้อมูล
2. reserveDA เป็นคลาสที่ใช้ควบคุมเพื่อการเข้าถึงข้อมูลความสนใจในการจองโครงการของลูกค้า
3. townhouseControl เป็นคลาสที่ใช้ควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภททาวน์เฮาส์ ไปยังระบบของเจ้าของโครงการ
4. condoControl เป็นคลาสที่ใช้ควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภทคอนโดมิเนียม ไปยังระบบของเจ้าของโครงการ
5. homeControl เป็นคลาสที่ใช้ในการควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภทบ้าน ไปยังระบบของเจ้าของโครงการ

6. login เป็นคลาสอินเทอร์เฟซที่ใช้ในการล็อกอินเข้าสู่ระบบ
7. editProfile เป็นคลาสอินเทอร์เฟซที่ใช้ในการแก้ไขข้อมูลของลูกค้า
8. register เป็นคลาสอินเทอร์เฟซที่ใช้ในการลงทะเบียนกับระบบ
9. reserve เป็นคลาสอินเทอร์เฟซที่ใช้ในการแจ้งความสนใจเพื่อจองโครงการ
10. townhouseSearch เป็นคลาสอินเทอร์เฟซที่ใช้ในการกำหนดเงื่อนไขเพื่อการค้นหาโครงการทาวน์เฮาส์
11. thcatelogResult เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการทาวน์เฮาส์ที่ตรงตามเงื่อนไข
12. townhouseDetail เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการทาวน์เฮาส์ในแต่ละโครงการ
13. townhouseModel เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลแบบทาวน์เฮาส์ในแต่ละแบบ
14. condoSearch เป็นคลาสอินเทอร์เฟซที่ใช้ในการกำหนดเงื่อนไขเพื่อการค้นหาโครงการคอนโดมิเนียม
15. codocatalogResult เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการคอนโดมิเนียมที่ตรงตามเงื่อนไข
16. condominiumDetail เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการคอนโดมิเนียมในแต่ละโครงการ
17. condoModel เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลแบบคอนโดมิเนียมในแต่ละแบบ
18. houseSearch เป็นคลาสอินเทอร์เฟซที่ใช้ในการกำหนดเงื่อนไขเพื่อการค้นหาโครงการบ้าน
19. catelogResult เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการบ้านที่ตรงตามเงื่อนไข
20. houseDetail เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลของโครงการบ้านในแต่ละโครงการ
21. houseModel เป็นคลาสอินเทอร์เฟซที่ใช้แสดงข้อมูลแบบบ้านในแต่ละแบบ
22. reservation เป็นคลาสเอนทิตีที่ใช้ในการเก็บรายละเอียดการแจ้งความสนใจ เพื่อจองโครงการและใช้ในการแปลงเป็นเอกสาร XML

3.1.5 ซีควেনซ์ไดอะแกรม

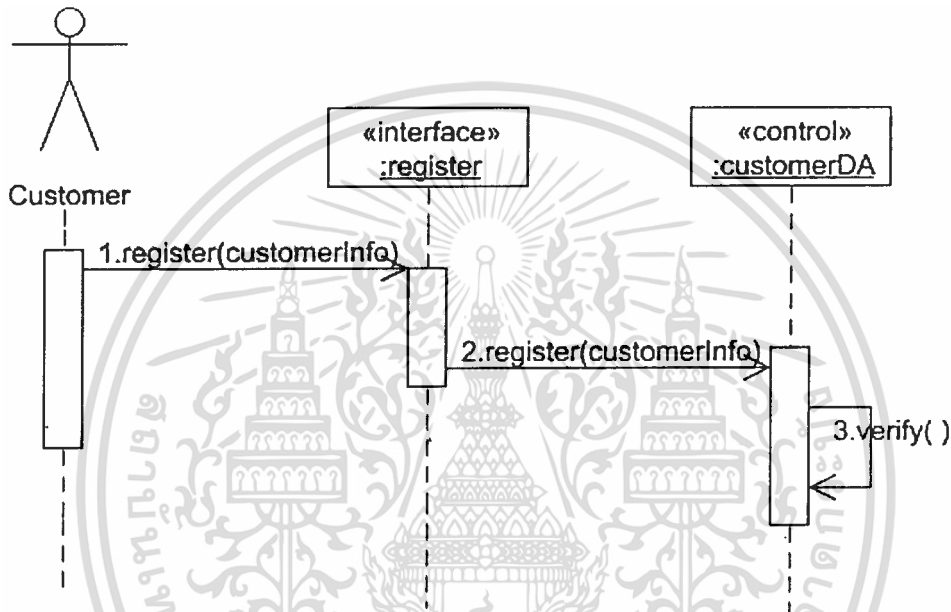
ซีควেনซ์ไดอะแกรมใช้อธิบายความสัมพันธ์ระหว่างยูสเคสและคลาสไดอะแกรม ซึ่ง

ประกอบไปด้วยส่วนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใด 3.1.4.1 ซีควেনซ์ไดอะแกรมของ register ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายการทำงานได้ดังนี้

1. ลูกค้าป้อนข้อมูลชื่อและที่อยู่ และregister จะทำการรับข้อมูลการลงทะเบียน
2. customerDA รับข้อมูลการลงทะเบียน
3. customerDA นำไปตรวจสอบว่าถูกต้องหรือไม่ ก่อนทำการบันทึกข้อมูลลงฐานข้อมูลของลูกค้า เมื่อลงทะเบียนเรียบร้อยแล้ว จะทำการบันทึกลงฐานข้อมูล

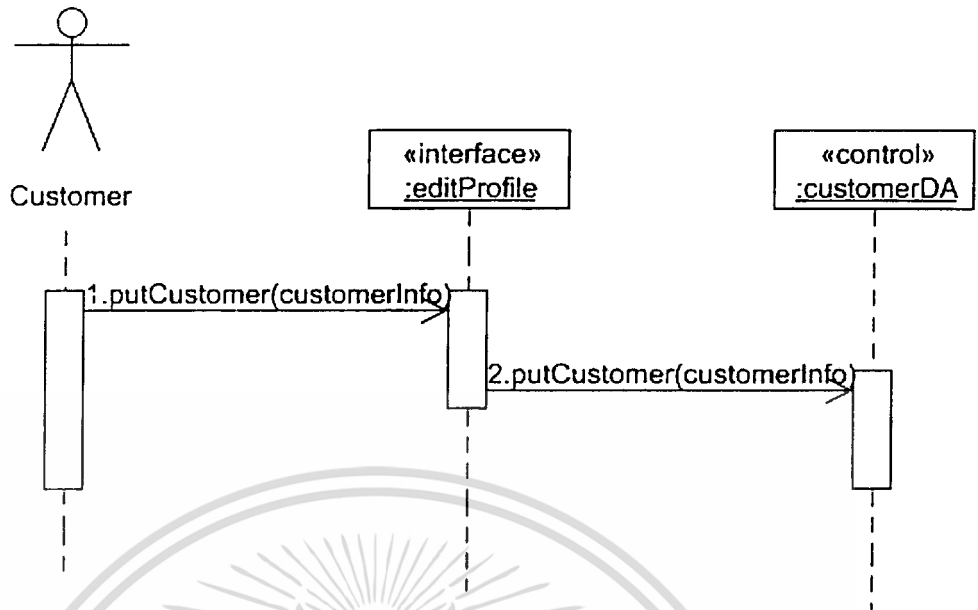


รูปที่ 3.4 ซีควেনซ์ไดอะแกรมของ register

3.1.4.2 ซีควেনซ์ไดอะแกรมของ Edit profile

อธิบายการทำงานได้ดังนี้

1. ลูกค้าทำการกรอกข้อมูลให้กับระบบ
2. customerDA ทำการรับข้อมูลและบันทึกข้อมูลใหม่

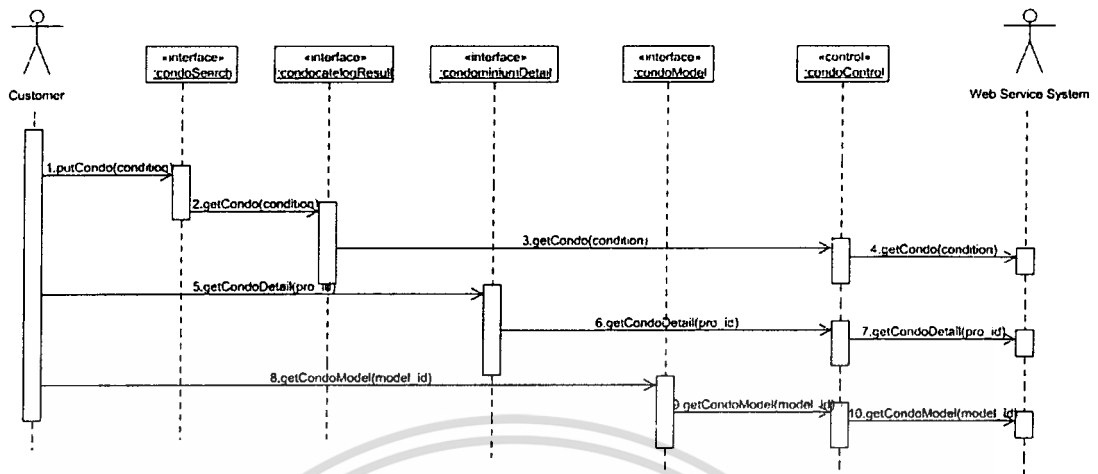


รูปที่ 3.5 ซีควেনซ์ไดอะแกรมของ Edit profile

3.1.4.3 ซีควেনซ์ไดอะแกรมของ view condominium

อธิบายการทำงานได้ดังนี้

1. ลูกค้าทำการเลือกและระบุเงื่อนไขที่ต้องการค้นหา
2. condocateLogResult ทำการรับข้อมูลและเงื่อนไขที่ส่งมาจาก condoSearch
3. condoControl รับข้อมูลและเงื่อนไขที่ส่งมาจาก condocateLogResult
4. condoControl ส่งข้อมูลและเงื่อนไขในการค้นหาไปยัง Web Service System
5. ลูกค้าทำการเลือกโครงการที่ต้องการทราบรายละเอียดไปให้กับ condominiumDetail
6. condominiumDetail ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง condoControl
7. condoControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System
8. ลูกค้าทำการเลือกแบบบ้านที่ต้องการทราบรายละเอียดไปให้กับ condoModel
9. condomodel ส่งรหัสแบบบ้านที่ต้องการทราบรายละเอียดไปยัง condoControl
10. condoControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System

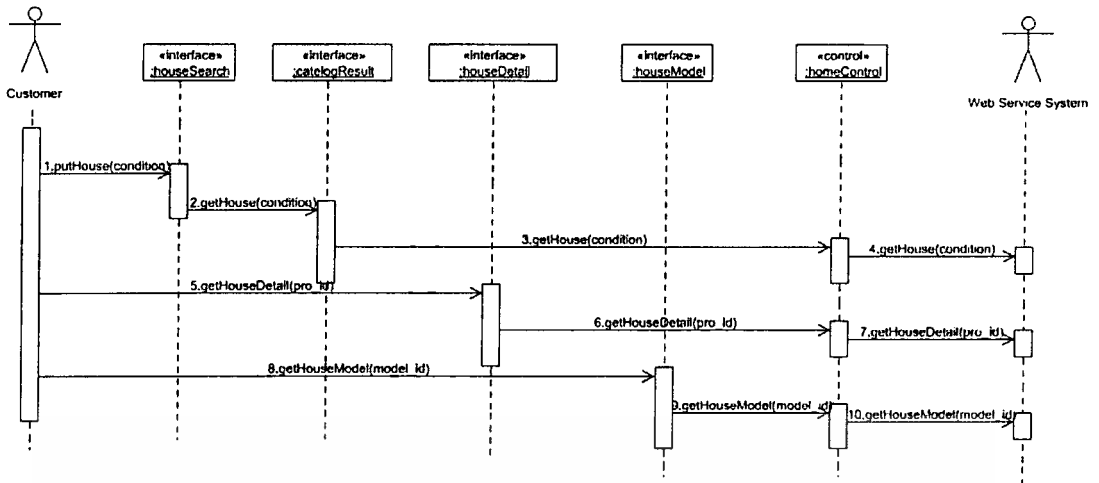


รูปที่ 3.6 ซีเควนซ์ไดอะแกรมของ view condominium

3.1.4.4 ซีเควนซ์ไดอะแกรมของ view house

อธิบายการทำงาน ได้ดังนี้

1. ลูกค้าทำการเลือกและระบุเงื่อนไขที่ต้องการค้นหา
2. catalogResult ทำการรับข้อมูลและเงื่อนไขที่ส่งมาจาก houseSearch
3. homeControl รับข้อมูลและเงื่อนไขที่ส่งมาจาก catalogResult
4. homeControl ส่งข้อมูลและเงื่อนไขในการค้นหาไปยัง Web Service System
5. ลูกค้าทำการเลือกโครงการที่ต้องการทราบรายละเอียดไปให้กับ houseDetail
6. houseDetail ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง homeControl
7. homeControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System
8. ลูกค้าทำการเลือกแบบบ้านที่ต้องการทราบรายละเอียดไปให้กับ houseModel
9. housemodel ส่งรหัสแบบบ้านที่ต้องการทราบรายละเอียดไปยัง homeControl
10. homeControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System

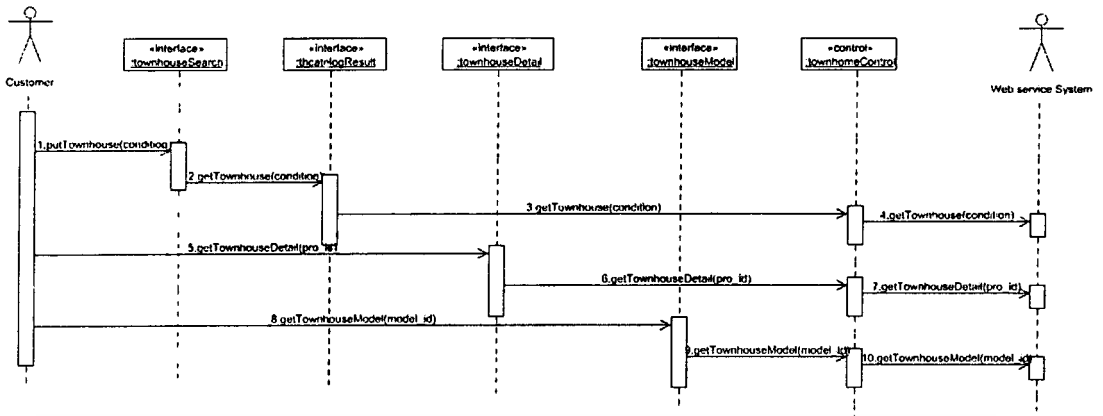


รูปที่ 3.7 ซีควেনซ์ไดอะแกรมของ view house

3.1.4.5 ซีควেনซ์ไดอะแกรมของ view townhouse

อธิบายการทำงานได้ดังนี้

1. ลูกค้าทำการเลือกและระบุเงื่อนไขที่ต้องการค้นหา
2. thcatalogResult ทำการรับข้อมูลและเงื่อนไขที่ส่งมาจาก townhouseSearch
3. townhouseControl รับข้อมูลและเงื่อนไขที่ส่งมาจาก thcatalogResult
4. townhouseControl ส่งข้อมูลและเงื่อนไขในการค้นหาไปยัง Web Service System
5. ลูกค้าทำการเลือกโครงการที่ต้องการทราบรายละเอียดไปให้กับ townhouseDetail
6. townhouseDetail ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง townhouseControl
7. townhouseControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System
8. ลูกค้าทำการเลือกแบบบ้านที่ต้องการทราบรายละเอียดไปให้กับ townhouseModel
9. townhousemodel ส่งรหัสแบบบ้านที่ต้องการทราบรายละเอียดไปยัง townhouseControl
10. townhouseControl ส่งรหัสโครงการที่ต้องการทราบรายละเอียดไปยัง Web Service System

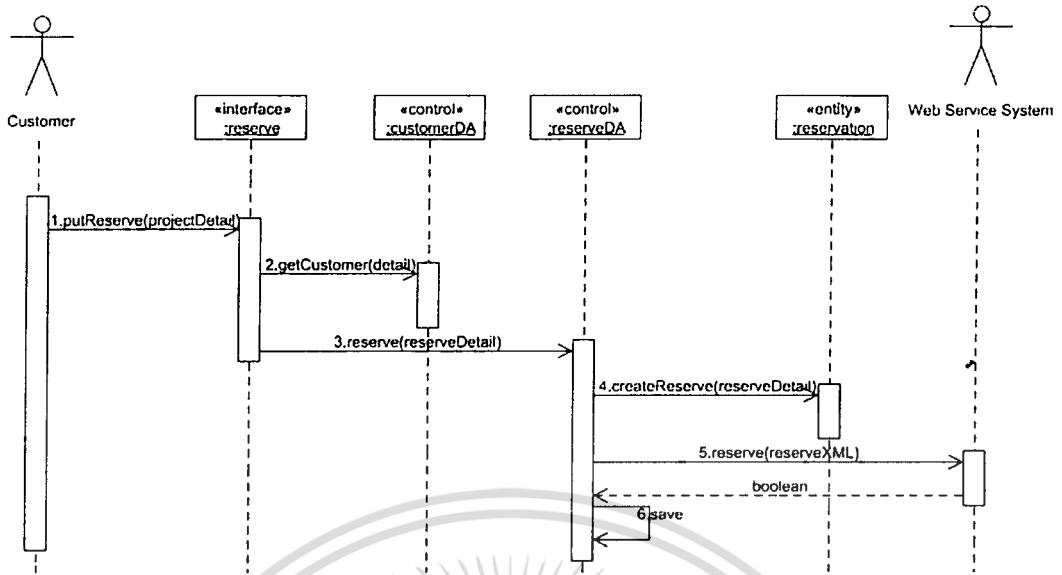


รูปที่ 3.8 ซีเควนซ์ไดอะแกรมของ view townhouse

3.1.4.6 ซีเควนซ์ไดอะแกรมของ request for reserve

อธิบายการทำงานได้ดังนี้

1. ลูกค้าล็อกอินเข้าสู่ระบบ ระบบโครงการที่สนใจ
2. customerDA ทำการรับข้อมูล username เพื่อทำการค้นหาข้อมูลลูกค้าแล้วส่งกลับไปให้ reserve
3. reserve ทำการส่งข้อมูลการจองไปยัง reserveDA
4. reserveDA ทำการสร้างอ็อบเจกต์ ตามรายละเอียดที่ได้รับแล้วแปลงจากอ็อบเจกต์เป็นเอกสาร XML
5. reserveDA ทำการส่งเอกสาร XML ไปยัง Web Service System จากนั้น Web Service System จะทำการส่งค่าตรรกะกลับมา ถ้าทำสำเร็จ boolean จะเป็น true ถ้าทำไม่สำเร็จ boolean จะเป็น false
6. ถ้า reserveDA ได้รับค่า boolean เป็น true ระบบจะทำการบันทึกรายการความสนใจโครงการของลูกค้าไว้เป็นหลักฐาน

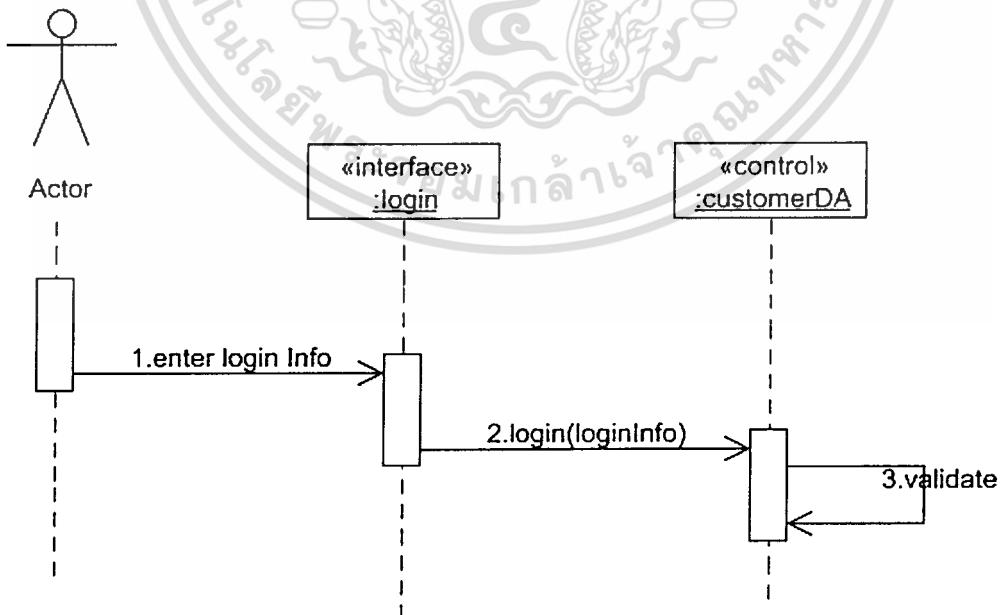


รูปที่ 3.9 ซีควเอนซ์ไดอะแกรมของ request for reserve

3.1.4.7 ซีควเอนซ์ไดอะแกรมของ validate account

อธิบายการทำงานได้ดังนี้

1. ลูกค้ากรอกข้อมูลเพื่อเข้าสู่ระบบ
2. login รับข้อมูลที่ลูกค้ากรอก และส่งไปยัง customerDA
3. customerDA ทำการตรวจสอบความถูกต้องของข้อมูลที่ลูกค้ากรอก



รูปที่ 3.10 ซีควเอนซ์ไดอะแกรมของ validate account

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 Real-estate1 Web Services และ Real-estate 2 Web Services

ระบบเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์ซึ่งเป็นเจ้าของโครงการรายที่ 1 และรายที่ 2 เป็นเว็บเซอร์วิสที่เป็นผู้ให้บริการ โดยจะรอรับการร้องขอ และส่งผลการตอบรับกลับไปยังผู้ร้องขอ (Central Web Service) ซึ่งมีการให้บริการดังนี้

- การค้นหาข้อมูลโครงการประเภทบ้านเดี่ยว
- การค้นหารายละเอียดแต่ละโครงการประเภทบ้านเดี่ยว
- การค้นหารายละเอียดแต่ละแบบบ้านในโครงการ
- การค้นหาข้อมูลโครงการประเภททาวเฮาส์
- การค้นหารายละเอียดแต่ละโครงการประเภททาวเฮาส์
- การค้นหารายละเอียดแต่ละแบบทาวเฮาส์ในโครงการ
- การค้นหาข้อมูลโครงการประเภทคอนโดมิเนียม
- การค้นหารายละเอียดแต่ละโครงการประเภทคอนโดมิเนียม
- การค้นหารายละเอียดแต่ละแบบคอนโดมิเนียมในโครงการ
- การแจ้งความสนใจโครงการ

การนำระบบดังกล่าวไปใช้งาน จำเป็นต้องมีการติดตั้งสภาพแวดล้อมดังนี้

1. Real-estate1 Web Services ติดตั้งอยู่บนระบบปฏิบัติการ Windows XP และระบบฐานข้อมูล MySQL
2. Real-estate 2 Web Services ติดตั้งอยู่บนระบบปฏิบัติการ Windows XP และระบบฐานข้อมูล Microsoft SQLserver 2000

3.3 การออกแบบระบบเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2

ระบบเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2 จะใช้สถาปัตยกรรมในการออกแบบระบบที่เหมือนกัน คือ ใช้เทคโนโลยี Java ซึ่งประกอบไปด้วย การร้องขอข้อมูลโครงการประเภทบ้านเดี่ยว การร้องขอข้อมูลโครงการประเภทคอนโดมิเนียม การร้องขอข้อมูลประเภททาวเฮาส์ การติดตามสถานะการจอง การส่งจองโครงการ แต่การออกแบบฐานข้อมูลของแต่ละบริษัทอสังหาริมทรัพย์จะแตกต่างกัน คือ บริษัทอสังหาริมทรัพย์รายที่ 1 จะใช้ฐานข้อมูล MySQL ส่วนบริษัทอสังหาริมทรัพย์รายที่ 2 จะใช้ฐานข้อมูล Microsoft SQLserver 2000

ขั้นตอนการทำงานของ บริษัทอสังหาริมทรัพย์จะถูกแบ่งออกเป็นหลายขั้นตอน ดังนี้

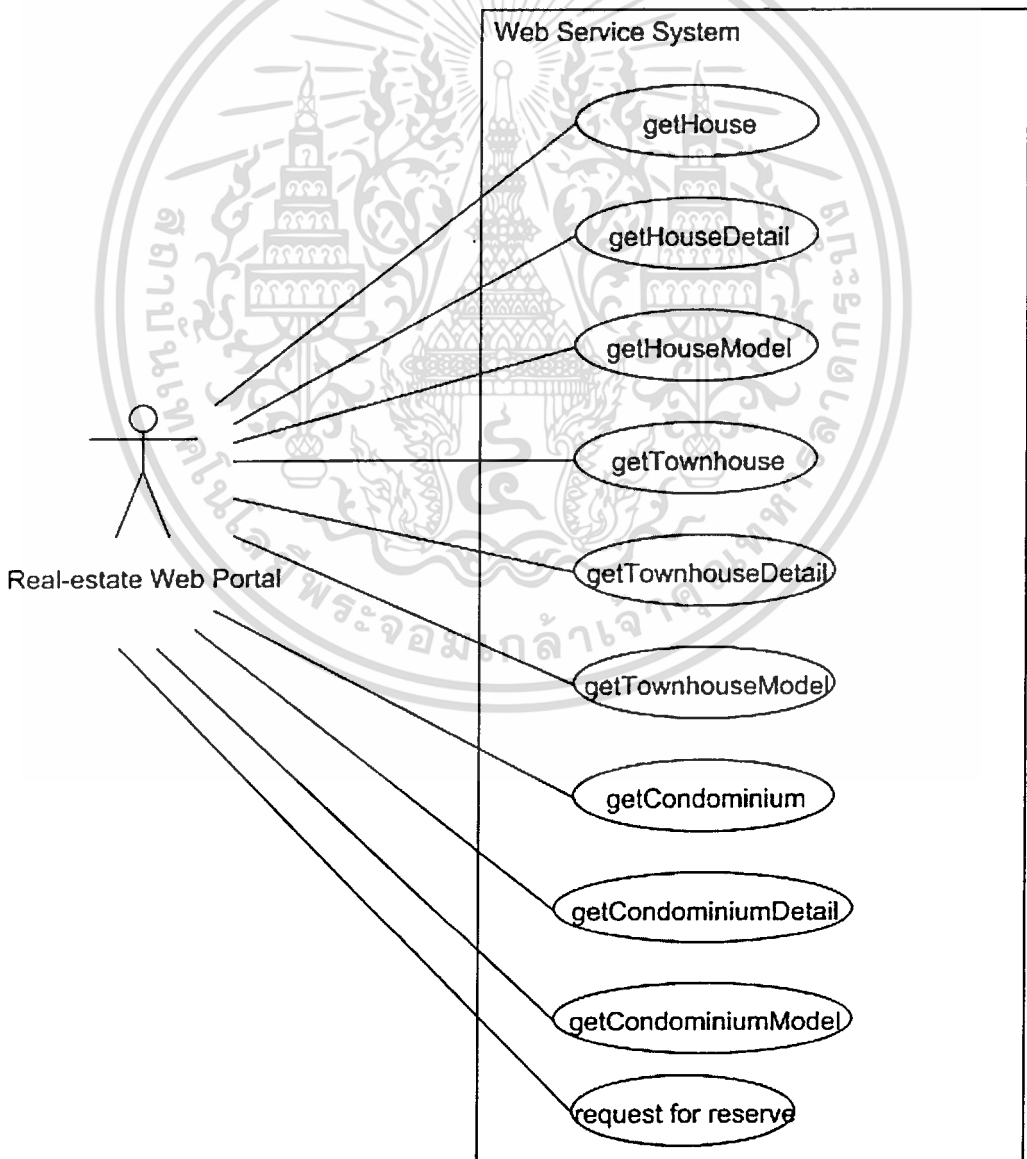
- 4 ผู้ดูแลระบบของบริษัทอสังหาริมทรัพย์จะใช้ JAXR ในการลงทะเบียนการให้บริการกับ

Registry Server

- 5 เมื่อลูกค้าต้องการเรียกใช้เซอร์วิสของบริษัทอสังหาริมทรัพย์ ลูกค้าจะทำการส่งข้อความเข้ามาตามรูปแบบที่แต่ละเซอร์วิสต้องการ
- 6 เมื่อผู้ดูแลระบบต้องการยกเลิกการให้บริการ ผู้ดูแลระบบจะใช้ JAXR ในการแจ้งยกเลิกการให้บริการกับ Registry Server

3.3.1 ยูสเคสไดอะแกรม

การวิเคราะห์และออกแบบระบบ เริ่มจากรวบรวมความต้องการ ซึ่งได้ 10 ยูสเคส คือ getHouse , getHouseDetail , getHouseModel , getTownhouse , getTownhouseDetail , getTownhouseModel , getCondominium , getCondominiumDetail , getCondominiumModel และ request for reserve และมี actor 1 actor คือ Real-estate Web Portal ดังรูปที่ 3.11



3.3.2 Use Case Description

ตารางที่ 3.9 รายละเอียดของยูสเคส getHouse

Use Case	getHouse
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายการโครงการบ้านโดยรวมที่ตรงตามเงื่อนไขการค้นหา
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดโครงการที่ตรงตามเงื่อนไขไปให้
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหารายการโครงการ
อินพุต	เงื่อนไขในการค้นหา
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งข้อมูลเพื่อใช้ในการค้นหาให้กับระบบ 2. ระบบทำการค้นหารายการโครงการที่ตรงตามเงื่อนไข 3. ส่งผลลัพธ์กลับไปที่ Real-estate Web Portal

ตารางที่ 3.10 รายละเอียดของยูสเคส getHouseDetail

Use Case	getHouseDetail
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายละเอียดแต่ละโครงการประเภทบ้าน
เงื่อนไขเมื่อเริ่มต้น	getHouse
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแต่ละโครงการที่ระบุไป
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ ทำการแจ้งความผิดพลาดไปให้
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาโครงการโดยรายละเอียด
อินพุต	รหัสโครงการบ้าน
เอาต์พุต	ผลลัพธ์ในการค้นหา

ตารางที่ 3.10 รายละเอียดของยูสเคส getHouseDetail (ต่อ)

รายละเอียด	1. Real-estate Web Portal ส่งรหัสโครงการบ้านให้กับระบบ 2. ระบบทำการค้นหาโครงการที่ระบุตามรหัสโครงการ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal
------------	---

ตารางที่ 3.11 รายละเอียดของยูสเคส getHouseModel

Use Case	getHouseModel
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายละเอียดแบบบ้านในโครงการ
เงื่อนไขเมื่อเริ่มต้น	getHouse
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแบบบ้านที่ระบุไป
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาแบบบ้านโดยรายละเอียด
อินพุต	รหัสแบบบ้าน
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งรหัสแบบบ้านให้กับระบบ 2. ระบบทำการค้นหาแบบบ้านตามรหัสที่ระบุ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.12 รายละเอียดของยูสเคส getTownhouse

Use Case	getTownhouse
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายการโครงการทาวน์เฮาส์โดยรวมที่ตรงตามเงื่อนไขการค้นหา
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดโครงการที่ตรงตามเงื่อนไขไปให้
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาโครงการ
อินพุต	เงื่อนไขในการค้นหา
เอาต์พุต	ผลลัพธ์ในการค้นหา

ตารางที่ 3.12 รายละเอียดของยูสเคส getTownhouse(ต่อ)

รายละเอียด	1. Real-estate Web Portal ส่งข้อมูลเพื่อใช้ในการค้นหาให้กับระบบ 2. ระบบทำการค้นหารายการโครงการที่ตรงตามเงื่อนไข 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal
------------	--

ตารางที่ 3.13 รายละเอียดของยูสเคส getTownhouseDetail

Use Case	getTownhouseDetail
วัตถุประสงค์	เพื่อค้นหารายละเอียดแต่ละโครงการประเภททาวน์เฮาส์
เงื่อนไขเมื่อเริ่มต้น	getTownhouse
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแต่ละโครงการที่ระบุไป
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาโครงการโดยรายละเอียด
อินพุต	รหัสโครงการทาวน์เฮาส์
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งรหัสโครงการทาวน์เฮาส์ให้กับระบบ 2. ระบบทำการโครงการที่ระบุไปตามรหัสโครงการ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.14 รายละเอียดของยูสเคส getTownhouseModel

Use Case	getTownhouseModel
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายละเอียดแบบทาวน์เฮาส์ในโครงการ
เงื่อนไขเมื่อเริ่มต้น	getTownhouseDetail
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแบบทาวน์เฮาส์ที่ระบุไป
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาแบบทาวน์เฮาส์โดยรายละเอียด

ตารางที่ 3.14 รายละเอียดของยูสเคส getTownhouseModel (ต่อ)

อินพุต	รหัสแบบทาวเฮาส์
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งรหัสแบบทาวเฮาส์ให้กับระบบ 2. ระบบทำการค้นหาแบบทาวเฮาส์ตามรหัสที่ระบุ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.15 รายละเอียดของยูสเคส getCondominium

Use Case	get Condominium
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายการ โครงการคอนโดมิเนียมโดยรวมที่ตรงตามเงื่อนไขการค้นหา
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดโครงการที่ตรงตามเงื่อนไขไปให้
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหารายการโครงการ
อินพุต	เงื่อนไขในการค้นหา
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งข้อมูลเพื่อใช้ในการค้นหาให้กับระบบ 2. ระบบทำการค้นหารายการโครงการที่ตรงตามเงื่อนไข 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.16 รายละเอียดของยูสเคส getCondominiumDetail

Use Case	getCondominiumDetail
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายละเอียดแต่ละ โครงการประเภทคอนโดมิเนียม
เงื่อนไขเมื่อเริ่มต้น	getCondominium
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแต่ละโครงการที่ระบุไป

ตารางที่ 3.16 รายละเอียดของยูสเคส getCondominiumDetail (ต่อ)

เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ ทำการแจ้งความผิดพลาดไปให้
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาโครงการโดยรายละเอียด
อินพุต	รหัสโครงการคอนโดมิเนียม
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งรหัสโครงการคอนโดมิเนียมให้กับระบบ 2. ระบบทำการโครงการที่ระบุไปตามรหัสโครงการ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.17 รายละเอียดของยูสเคส getCondominiumModel

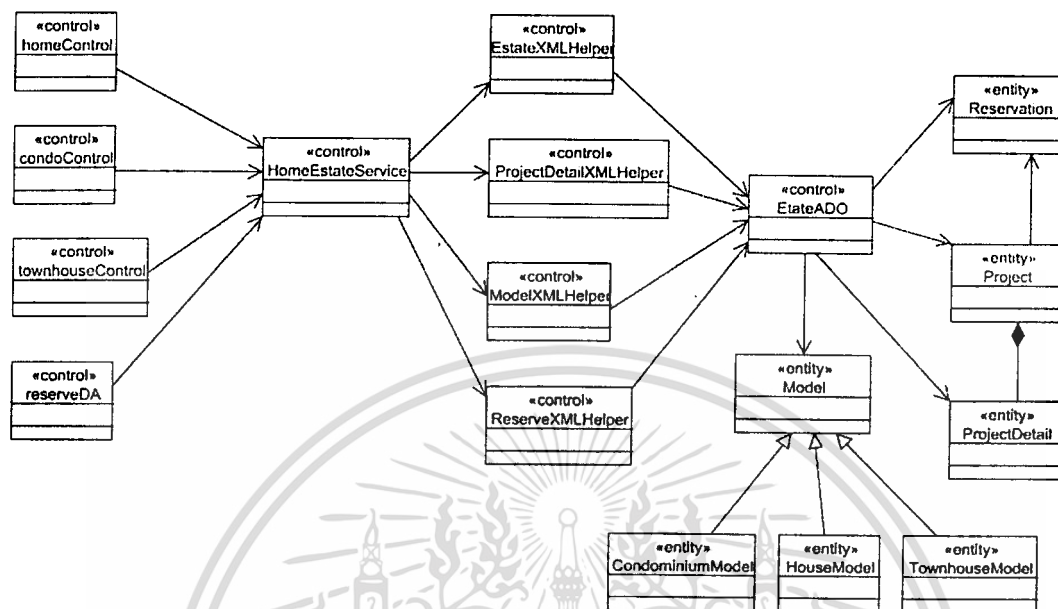
Use Case	getCondominiumModel
วัตถุประสงค์	เป็นบริการเพื่อค้นหารายละเอียดแบบคอนโดมิเนียมในโครงการ
เงื่อนไขเมื่อเริ่มต้น	getCondominiumDetail
เมื่องานทำสำเร็จ	ระบบแสดงรายละเอียดแบบคอนโดมิเนียมที่ระบุไป
เมื่อทำงานไม่สำเร็จ	ระบบหาข้อมูลไม่เจอ
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal เรียกเมธอดในการค้นหาแบบคอนโดมิเนียมโดยรายละเอียด
อินพุต	รหัสแบบคอนโดมิเนียม
เอาต์พุต	ผลลัพธ์ในการค้นหา
รายละเอียด	1. Real-estate Web Portal ส่งรหัสแบบคอนโดมิเนียมให้กับระบบ 2. ระบบทำการค้นหาแบบคอนโดมิเนียมตามรหัสที่ระบุ 3. ส่งผลลัพธ์กลับไปให้ Real-estate Web Portal

ตารางที่ 3.18 รายละเอียดของยูสเคส request for reserve

Use Case	request for reserve
วัตถุประสงค์	เป็นบริการที่ให้ลูกค้าใช้ในการแจ้งความสนใจโครงการเพื่อการจองในลำดับต่อไป
เงื่อนไขเมื่อเริ่มต้น	-
เมื่องานทำสำเร็จ	ระบบบันทึกข้อมูลการร้องขอดังกล่าว และReal-estate Web Portal จะได้รับข้อมูลการตอบรับเป็นตรรกะ true
เมื่อทำงานไม่สำเร็จ	Real-estate Web Portal ได้รับการตอบกลับเป็นตรรกะ false
Actor ที่เกี่ยวข้อง	Real-estate Web Portal
สิ่งกระตุ้นการทำงาน	Real-estate Web Portal ทำการเรียกเมทอด reserve
อินพุต	ข้อมูลการแจ้งความจำนงในรูปแบบ XML
เอาต์พุต	ค่าตรรกะ
รายละเอียด	1 Real-estate Web Portal ทำการเรียกเมทอด reserve 2 ระบบทำการบันทึกข้อมูลแล้วให้ค่าตรรกะเป็น true 3 ระบบทำการส่งค่าตรรกะกลับไปให้ Real-estate Web Portal

3.3.3 คลาสไดอะแกรม

สำหรับรายละเอียดของคลาสภายใต้โปรแกรมที่ต้องการสร้างขึ้น และความสัมพันธ์ต่างๆ ระหว่างคลาส โดยคลาสใช้สัญลักษณ์เป็นสี่เหลี่ยมผืนผ้า ภายในแบ่งเป็น ชื่อคลาส คุณสมบัติ และ เมทอดเรียงจากบนลงล่าง ดังรูปที่ 3.12



รูปที่ 3.12 คลาสไดอะแกรมของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์

คลาสภายในโปรแกรมที่ต้องสร้างขึ้น ได้แก่

1. homeControl เป็นคลาสที่ใช้ในการควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภทบ้าน ไปยังระบบของเจ้าของโครงการ
2. condoControl เป็นคลาสที่ใช้ควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภทคอนโดมิเนียม ไปยังระบบของเจ้าของโครงการ
3. townhouseControl เป็นคลาสที่ใช้ควบคุมเพื่อการค้นหาและแจ้งความสนใจในการจองโครงการของโครงการประเภททาวน์เฮาส์ ไปยังระบบของเจ้าของโครงการ
4. reserveDA เป็นคลาสที่ใช้ควบคุมเพื่อการเข้าถึงข้อมูลความสนใจในการจองโครงการของลูกค้า
5. HomeEstateService เป็นคลาสที่ใช้ควบคุมการบริการทั้งหมดของระบบ
6. EstateXMLHelper เป็นคลาสที่ใช้ควบคุมการแปลงข้อมูลจากอ็อบเจกต์เป็น XML และจาก XML เป็นอ็อบเจกต์ของโครงการ
7. ProjectDetailXMLHelper เป็นคลาสที่ใช้ควบคุมการแปลงข้อมูลจากอ็อบเจกต์เป็น XML และจาก XML เป็นอ็อบเจกต์ของแต่ละโครงการ
8. ModelXMLHelper เป็นคลาสที่ใช้ควบคุมการแปลงข้อมูลจากอ็อบเจกต์เป็น XML และจาก XML เป็นอ็อบเจกต์ของแบบบ้าน ทาวน์เฮาส์ และคอนโดมิเนียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

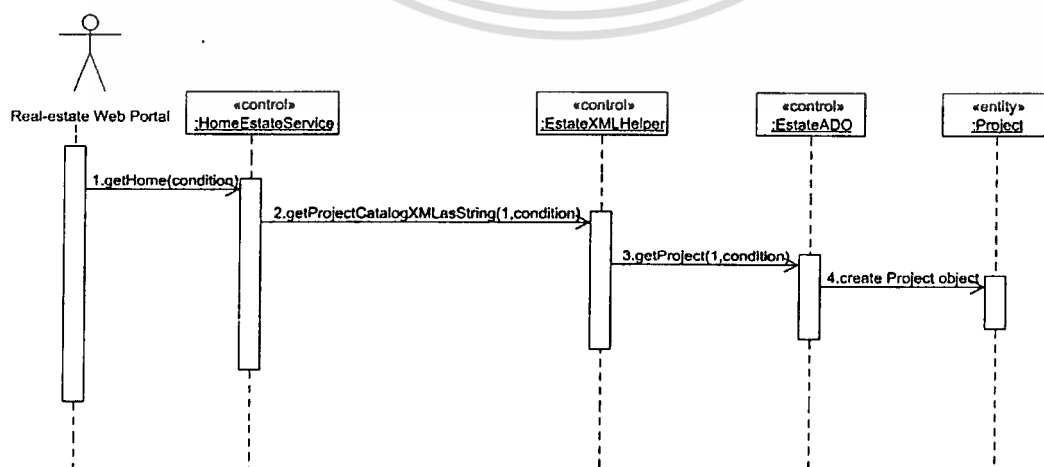
9. ReserveXMLHelper เป็นคลาสที่ใช้ควบคุมการแปลงข้อมูลจากอ็อบเจ็กเป็น XML และจาก XML เป็นอ็อบเจ็ก ของรายการแจ้งความสนใจโครงการ
10. EstateADO เป็นคลาสที่ใช้ควบคุมในการติดต่อกับฐานข้อมูล
11. Reservation เป็นคลาสสำหรับใช้เก็บรายละเอียดรายการแจ้งความสนใจโครงการ
12. Project เป็นคลาสสำหรับเก็บข้อมูลของโครงการทุกประเภท
13. ProjectDetail เป็นคลาสที่ใช้เก็บข้อมูลแต่ละโครงการ
14. Model เป็นคลาสที่ใช้เก็บแบบก่อสร้างหิรัญทรัพย์ทุกประเภท
15. CondominiumModel เป็นคลาสสำหรับเก็บรายละเอียดแบบของคอนโดมิเนียม
16. HouseModel เป็นคลาสสำหรับเก็บรายละเอียดแบบบ้าน
17. TownhouseModel เป็นคลาสสำหรับเก็บรายละเอียดแบบทาวน์เฮาส์

3.3.4 ซีควেনซ์ไดอะแกรม

3.3.4.1 ซีควেনซ์ไดอะแกรมของ getHouse

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งเงื่อนไขของโครงการบ้านที่ต้องการค้นหาให้กับ HomeEstateService
2. EstateXMLHelper ทำการรับเงื่อนไขและประเภทประเภทของโครงการบ้าน ที่ส่งมาจาก HomeEstateService
3. EstateADO รับเงื่อนไขในการค้นหาและประเภทของโครงการ(ในที่นี้ กำหนดให้เป็น 1 ซึ่งหมายถึงโครงการประเภทบ้าน)
4. EstateADO ทำการสร้างอ็อบเจ็กโครงการบ้านที่ตรงตามเงื่อนไขแล้วส่งกลับไป

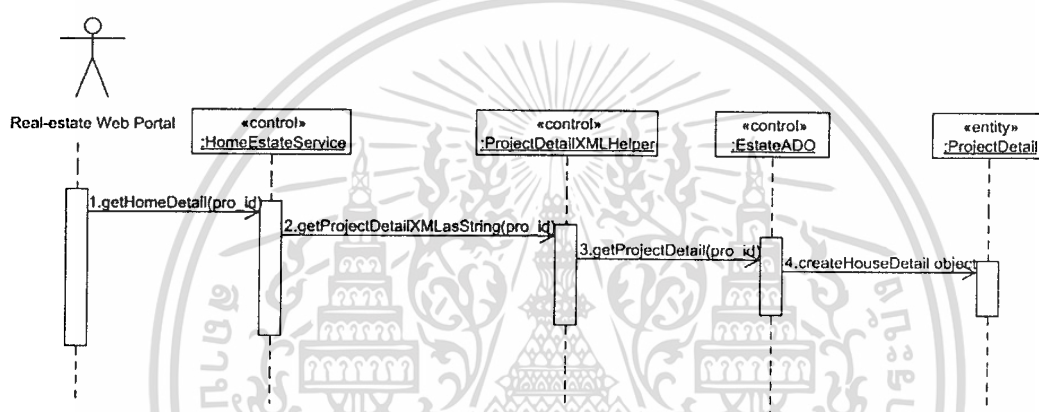


เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และใช้เฉพาะเพื่อการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4.2 ซีควเอนซ์ไดอะแกรมของ getHouseDetail

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ส่ง id ของโครงการบ้านที่ต้องการค้นหาในรายละเอียด
2. ProjectDetailXMLHelper ทำการรับ id ของโครงการบ้านที่ต้องการค้นหาในรายละเอียด ที่ส่งมาจาก HomeEstateService
3. EstateADO ทำการรับ pro_id ซึ่งเป็นรหัสโครงการ ที่ส่งมาจาก ProjectDetailXMLHelper
4. EstateADO ทำการสร้างอ็อบเจ็กต์ ProjectDetail แล้วส่งกลับไป

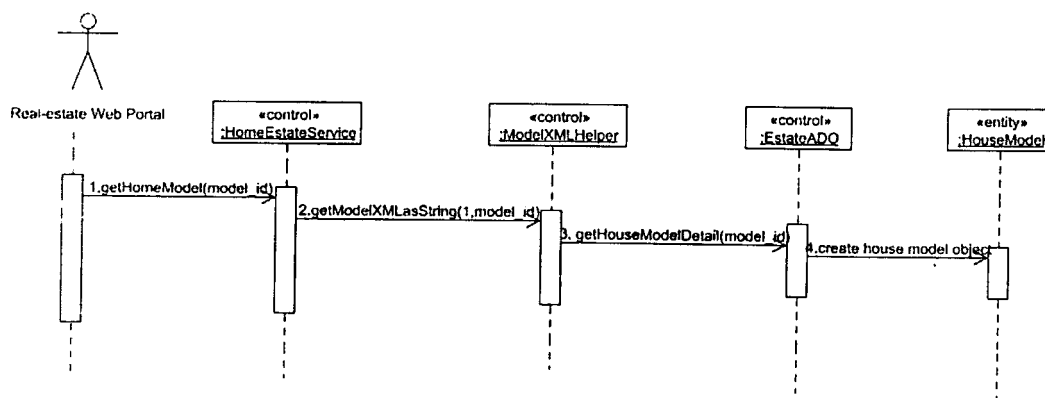


รูปที่ 3.14 ซีควเอนซ์ไดอะแกรมของ getHouseDetail

3.3.4.3 ซีควเอนซ์ไดอะแกรมของ getHouseModel

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการสร้างรหัสแบบบ้านที่ต้องการค้นหาให้กับ HomeEstateService
2. ModelXMLHelper ทำการรับประเภทโครงการและ รหัสแบบบ้านที่ส่งมาจาก HomeEstateService
3. EstateADO รับรหัสแบบบ้านเพื่อใช้ในการค้นหาข้อมูลแบบบ้าน
4. EstateADO ทำการสร้างอ็อบเจ็กต์ HouseModel แล้วส่งกลับไป

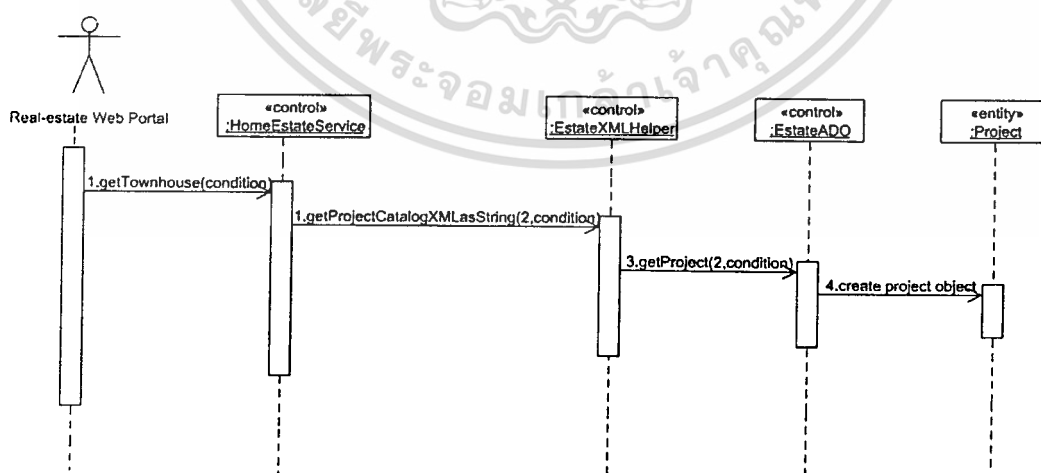


รูปที่ 3.15 ซีควেনซ์ไดอะแกรมของ getHouseModel

3.3.4.4 ซีควেনซ์ไดอะแกรมของ getTownhouse

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งเงื่อนไขของโครงการทาวน์เฮาส์ที่ต้องการค้นหาให้กับ HomeEstateService
2. EstateXMLHelper ทำการรับเงื่อนไขและประเภทประเภทของโครงการทาวน์เฮาส์ ที่ส่งมาจาก HomeEstateService
3. EstateADO รับเงื่อนไขในการค้นหาและประเภทของโครงการ(ในที่นี้ กำหนดให้เป็น 2 ซึ่งหมายถึงโครงการประเภททาวน์เฮาส์)
4. EstateADO ทำการสร้างอ็อบเจกต์โครงการทาวน์เฮาส์ที่ตรงตามเงื่อนไขแล้วส่งกลับไป



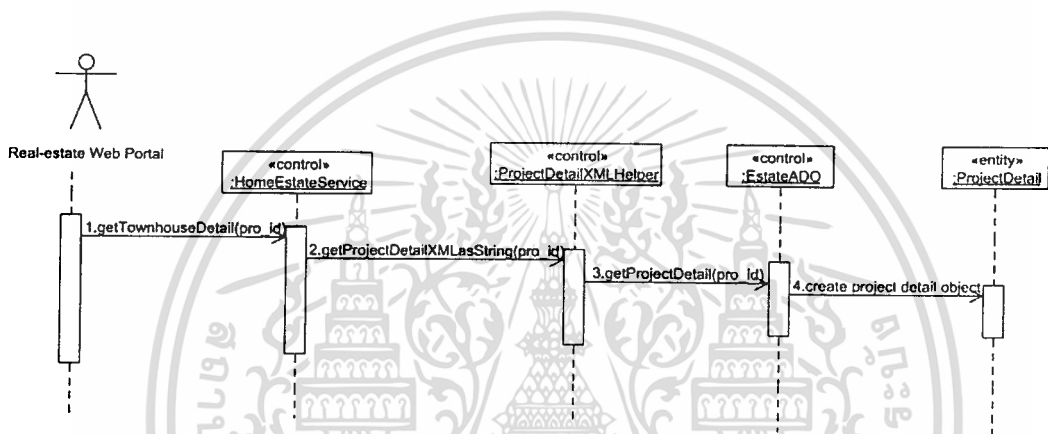
รูปที่ 3.16 ซีควেনซ์ไดอะแกรมของ getTownhouse

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4.5 ซีควเอนซ์ไดอะแกรมของ getTownhouseDetail

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ส่ง id ของโครงการทาวเฮาส์ที่ต้องการค้นหาในรายละเอียด
2. ProjectDetailXMLHelper ทำการรับ id ของโครงการทาวเฮาส์ที่ต้องการค้นหาในรายละเอียด ที่ส่งมาจาก HomeEstateService
3. EstateADO ทำการรับ pro_id ซึ่งเป็นรหัสโครงการ ที่ส่งมาจาก ProjectDetailXMLHelper
4. EstateADO ทำการสร้างอ็อบเจกต์ ProjectDetail แล้วส่งกลับไป

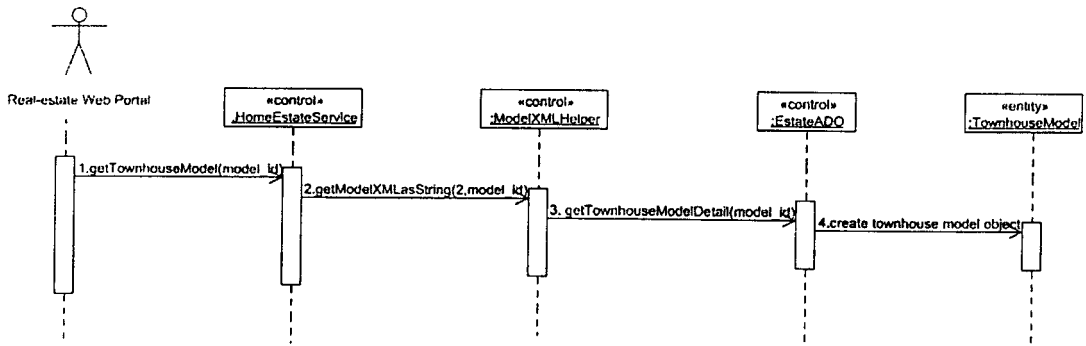


รูปที่ 3.17 ซีควเอนซ์ไดอะแกรมของ getTownhouseDetail

3.3.4.6 ซีควเอนซ์ไดอะแกรมของ getTownhouseModel

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งรหัสแบบทาวเฮาส์ที่ต้องการค้นหาให้กับ HomeEstateService
2. ModelXMLHelper ทำการรับประเภทโครงการและ รหัสแบบทาวเฮาส์ที่ส่งมาจาก HomeEstateService
3. EstateADO รับรหัสแบบทาวเฮาส์เพื่อใช้ในการค้นหาข้อมูลแบบทาวเฮาส์
4. EstateADO ทำการสร้างอ็อบเจกต์ TownhouseModel แล้วส่งกลับไป

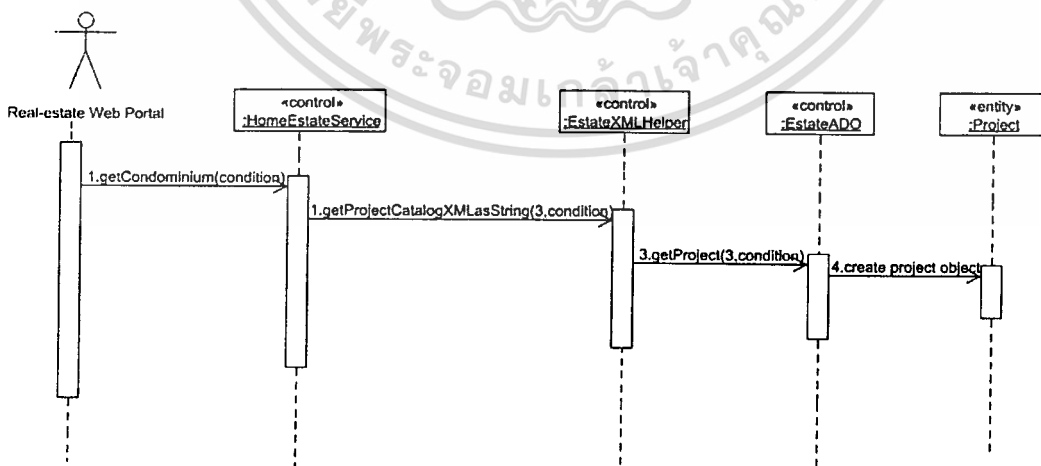


รูปที่ 3.18 ซีควেনซ์ไดอะแกรมของ getTownhouseModel

3.3.4.7 ซีควেনซ์ไดอะแกรมของ getCondominium

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งเงื่อนไขของโครงการคอนโดมิเนียมที่ต้องการค้นหาให้กับ HomeEstateService
2. EstateXMLHelper ทำการรับเงื่อนไขและประเภทประเภทของโครงการคอนโดมิเนียม ที่ส่งมาจาก HomeEstateService
3. EstateADO รับเงื่อนไขในการค้นหาและประเภทของโครงการ(ในที่นี้ กำหนดให้เป็น 3 ซึ่งหมายถึงโครงการประเภทคอนโดมิเนียม)
4. EstateADO ทำการสร้างอ็อบเจกต์โครงการคอนโดมิเนียมที่ตรงตามเงื่อนไขแล้วส่งกลับไป



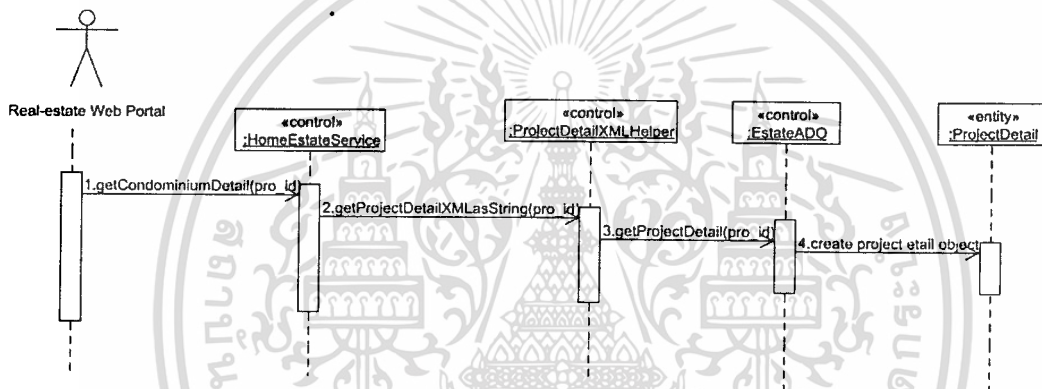
รูปที่ 3.19 ซีควেনซ์ไดอะแกรมของ getCondominium

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4.8 ซีเควนซ์ไดอะแกรมของ getCondominiumDetail

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ส่ง id ของโครงการคอนโดมิเนียมที่ต้องการค้นหาในรายละเอียด
2. ProjectDetailXMLHelper ทำการรับ id ของโครงการคอนโดมิเนียมที่ต้องการค้นหาในรายละเอียด ที่ส่งมาจาก HomeEstateService
3. EstateADO ทำการรับ pro_id ซึ่งเป็นรหัสโครงการ ที่ส่งมาจาก ProjectDetailXMLHelper
4. EstateADO ทำการสร้างอ็อบเจ็กต์ ProjectDetail แล้วส่งกลับไป

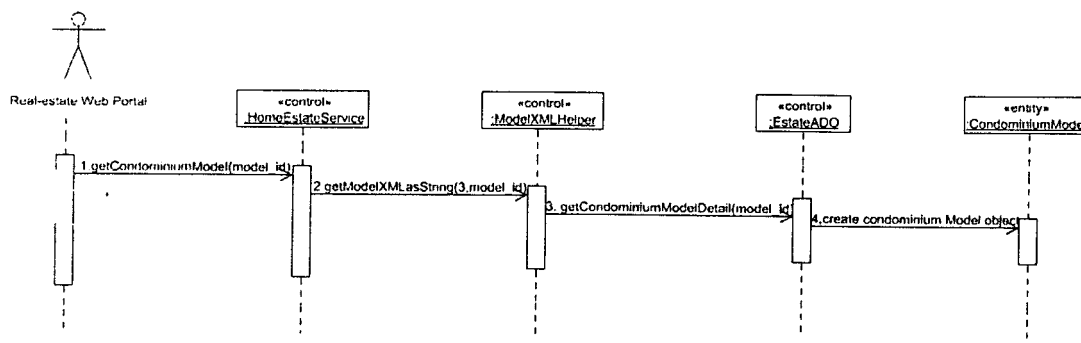


รูปที่ 3.20 ซีเควนซ์ไดอะแกรมของ getComdominiumDetail

3.3.4.9 ซีเควนซ์ไดอะแกรมของ getCondominiumModel

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งรหัสแบบคอนโดมิเนียมที่ต้องการค้นหาให้กับ HomeEstateService
2. ModelXMLHelper ทำการรับประเภทโครงการและ รหัสแบบคอนโดมิเนียมที่ส่งมาจาก HomeEstateService
3. EstateADO รับรหัสแบบคอนโดมิเนียมเพื่อใช้ในการค้นหาข้อมูลแบบคอนโดมิเนียม
4. EstateADO ทำการสร้างอ็อบเจ็กต์ CondominiumModel แล้วส่งกลับไป

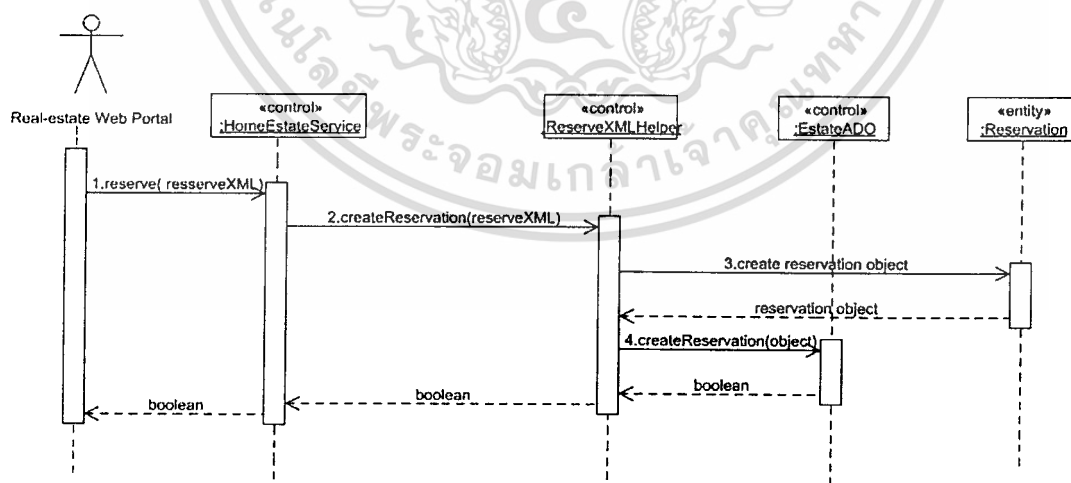


รูปที่ 3.21 ซีเควนซ์ไดอะแกรมของ getCondominiumModel

3.3.4.10 ซีเควนซ์ไดอะแกรมของ request for reserve

อธิบายการทำงานได้ดังนี้

1. Real-estate Web Portal ทำการส่งการแจ้งความสนใจโครงการในรูปแบบเอกสาร XML มายัง HomeEstateService
2. ReserveXMLHelper ทำการรับเอกสาร XML ที่ส่งมาจาก HomeEstateService
3. ReserveXMLHelper ทำการสร้างอ็อบเจ็กต์ Reservation
4. ReserveXMLHelper ทำการส่งอ็อบเจ็กต์ Reservation ไปยัง EstateADO เพื่อทำการบันทึกต่อไป จากนั้น EstateADO จะส่งค่าตรรกะกลับไป



รูปที่ 3.22 ซีเควนซ์ไดอะแกรมของ request for reserve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

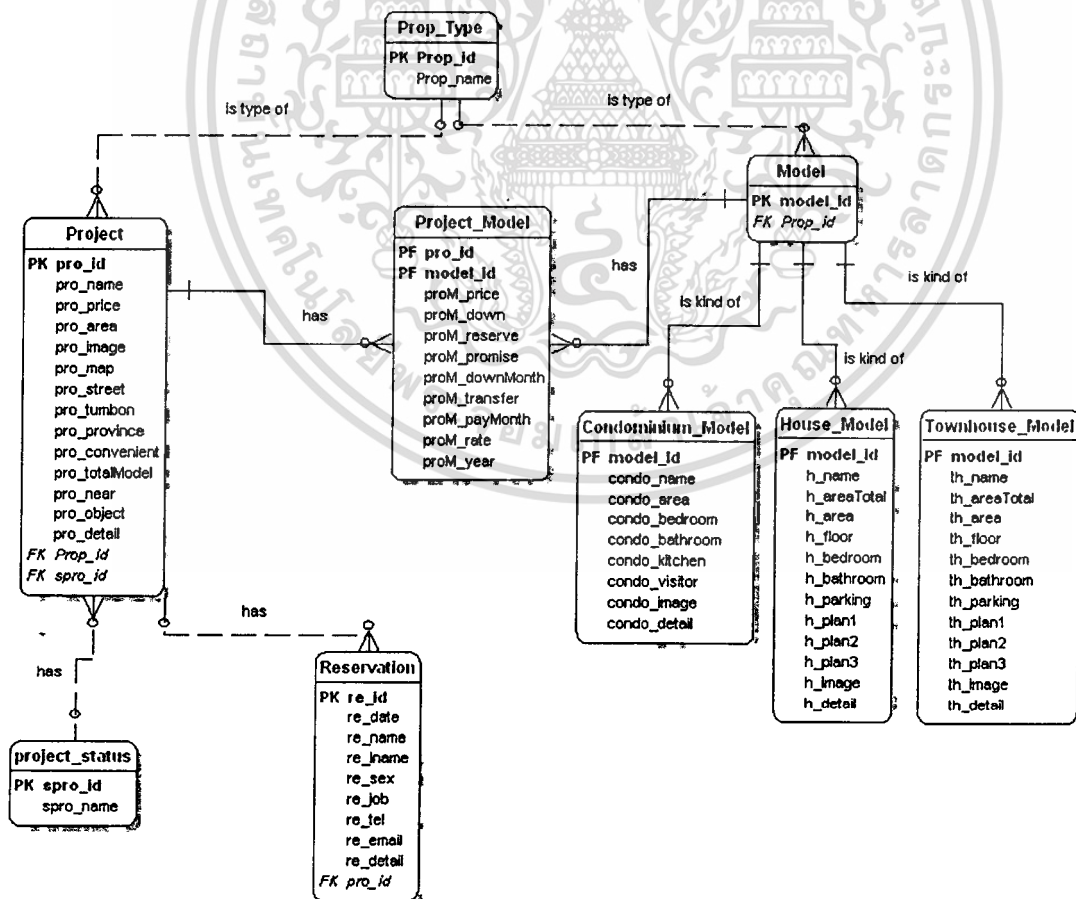
บทที่ 4

การออกแบบฐานข้อมูล

โครงการนี้มีระบบที่ใช้ติดต่อกับฐานข้อมูลอยู่ 3 ระบบ คือ ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 2 โดยที่ฐานข้อมูลทั้ง 2 จะออกแบบมาให้เหมือนกัน และฐานข้อมูลของเว็บกลาง โดยมีรายละเอียดดังต่อไปนี้

4.1 ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2

ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 จะพัฒนาด้วย MySQL และฐานข้อมูลฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 2 จะพัฒนาด้วย Microsoft SQLserver 2000 ประกอบด้วย 10 ตารางดังรูปที่ 4.1



รูปที่ 4.1 ER Diagram ฐานข้อมูลของเว็บเซอร์วิสของบริษัทอสังหาริมทรัพย์รายที่ 1 และ 2

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำเอกสารนี้ไปใช้ในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.1 มีรายละเอียดดังนี้

- 1. Reservation เป็นตารางที่ใช้เก็บรายละเอียดการจอง โดยรายละเอียดแสดงดังตารางที่ 4.1
- 2. Project เป็นตารางที่ใช้เก็บรายละเอียดของโครงการ โดยรายละเอียดแสดงดังตารางที่ 4.2
- 3. Prop_Type เป็นตารางที่ใช้เก็บประเภทของอสังหาริมทรัพย์ โดยรายละเอียดแสดงดังตารางที่ 4.3
- 4. Model เป็นตารางที่ใช้เก็บรายละเอียดของแบบอสังหาริมทรัพย์ทุกประเภท โดยรายละเอียดแสดงดังตารางที่ 4.4
- 5. Project_Model เป็นตารางที่ใช้เก็บรายละเอียดว่าแต่ละโครงการมีแบบอสังหาริมทรัพย์อะไรบ้าง และมีค่าใช้จ่ายเท่าไรบ้าง โดยรายละเอียดแสดงดังตารางที่ 4.5
- 6. Condominium_Model เป็นตารางที่ใช้เก็บรายละเอียดของแบบคอนโดมิเนียม โดยรายละเอียดแสดงดังตารางที่ 4.6
- 7. House_Model เป็นตารางที่ใช้เก็บรายละเอียดของแบบบ้าน โดยรายละเอียดแสดงดังตารางที่ 4.7
- 8. Townhouse เป็นตารางที่ใช้เก็บรายละเอียดของแบบทาวน์เฮาส์ โดยรายละเอียดแสดงดังตารางที่ 4.8
- 9. project_status เป็นตารางที่ใช้เก็บรายละเอียดของสถานะโครงการ โดยรายละเอียดแสดงดังตารางที่ 4.9

ตารางที่ 4.1 รายละเอียดฐานข้อมูลของตาราง Reservation

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
re_id	INTEGER	Primary Key		รหัสการจอง
re_date	DATE	-		วันที่จอง
re_name	VARCHAR(50)	-		ชื่อผู้แจ้งความสนใจ
re_lname	VARCHAR(50)	-		นามสกุลผู้แจ้งความสนใจ
re_sex	VARCHAR(10)	-		เพศ
re_job	VARCHAR(30)	-		อาชีพ
re_tel	VARCHAR(20)	-		เบอร์โทรศัพท์
re_email	VARCHAR(30)	-		อีเมลผู้แจ้งความ งาน

ตารางที่ 4.1 รายละเอียดฐานข้อมูลของตาราง Reservation (ต่อ)

re_detail	VARCHAR(200)	-		รายละเอียดอื่นๆ
pro_id	INTEGER	-		รหัสโครงการ

ตารางที่ 4.2 รายละเอียดฐานข้อมูลของตาราง Project

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
pro_id	INTEGER	Primary Key		รหัสโครงการ
pro_name	VARCHAR(40)	-		ชื่อโครงการ
pro_price	FLOAT	-		ราคาเริ่มต้น
pro_area	INTEGER	-		พื้นที่โครงการ
pro_image	VARCHAR(100)	-		รูปภาพโครงการ
pro_map	VARCHAR(100)	-		แผนที่โครงการ
pro_street	VARCHAR(30)	-		ถนน
pro_tumbon	VARCHAR(30)	-		เขต
pro_province	VARCHAR(30)	-		จังหวัด
pro_convenient	VARCHAR(300)	-		สิ่งอำนวยความสะดวก
pro_totalModel	INTEGER	-		จำนวนแบบ
pro_near	VARCHAR(500)	-		สถานที่แวดล้อม
pro_object	VARCHAR(500)	-		วัตถุประสงค์โครงการ
pro_detail	VARCHAR(1000)	-		รายละเอียดโครงการ
prop_id	INTEGER	Foreign Key	Prop_Type	รหัส อสังหาริมทรัพย์
spro_id	INTEGER	Foreign Key	project_status	รหัสสถานะโครงการ

ตารางที่ 4.3 รายละเอียดฐานข้อมูลของตาราง Prop_Type

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
prop_id	INTEGER	Primary Key		รหัสประเภทโครงการ

ตารางที่ 4.3 รายละเอียดฐานข้อมูลของตาราง Prop_Type(ต่อ)

prop_name	VARCHAR(30)	-		ชื่อประเภทโครงการ
-----------	-------------	---	--	-------------------

ตารางที่ 4.4 รายละเอียดฐานข้อมูลของตาราง Model

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
model_id	INTEGER	Primary Key		รหัสแบบ อสังหาริมทรัพย์
prop_id	INTEGER	Foreign Key	Prop_Type	รหัสประเภท โครงการ

ตารางที่ 4.5 รายละเอียดฐานข้อมูลของตาราง Project_Model

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
pro_id	INTEGER	Primary Key, Foreign Key	Project	รหัสโครงการ
model_id	INTEGER	Primary Key, Foreign Key	Model	รหัสแบบ อสังหาริมทรัพย์
proM_price	VARCHAR(40)	-		ราคาเริ่มต้น
proM_down	VARCHAR(40)	-		ราคาคาวน
proM_reserve	VARCHAR(40)	-		ราคาจอง
proM_promise	VARCHAR(40)	-		ค่าทำสัญญา
proM_downMonth	VARCHAR(40)	-		ราคาคาวนต่อเดือน
proM_transfer	VARCHAR(40)	-		ราคาโอน
proM_payMonth	VARCHAR(40)	-		ราคาจ่ายต่อเดือน
proM_rate	VARCHAR(40)	-		อัตราดอกเบี้ย
proM_year	VARCHAR(40)	-		จำนวนปี

ตารางที่ 4.6 รายละเอียดฐานข้อมูลของตาราง Condominium_Model

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
model_id	INTEGER	Primary Key, Foreign Key	Model	รหัสแบบ อสังหาริมทรัพย์
condo_name	VARCHAR(30)	-		ชื่อแบบคอนโด
condo_area	VARCHAR(30)	-		พื้นที่คอนโด

ตารางที่ 4.6 รายละเอียดฐานข้อมูลของตาราง Condominium_Model (ต่อ)

condo_bedroom	INTEGER	-		จำนวนห้องนอน
condo_bathroom	INTEGER	-		จำนวนห้องน้ำ
condo_kitchen	INTEGER	-		จำนวนห้องครัว
condo_visitor	INTEGER	-		จำนวนห้องรับแขก
condo_image	VARCHAR(100)	-		ภาพแบบคอนโด
condo_detail	VARCHAR(1000)	-		รายละเอียดคอนโด

ตารางที่ 4.7 รายละเอียดฐานข้อมูลของตาราง House_Model

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
model_id	INTEGER	Primary Key, Foreign Key	Model	รหัสแบบ อสังหาริมทรัพย์
h_name	VARCHAR(30)	-		ชื่อแบบบ้าน
h_area	VARCHAR(30)	-		พื้นที่ บ้าน
h_floor	INTEGER	-		จำนวนชั้น
h_bedroom	INTEGER	-		จำนวนห้องนอน
h_bathroom	INTEGER	-		จำนวนห้องน้ำ
h_parking	INTEGER	-		จำนวนที่จอดรถ
h_plan1	VARCHAR(100)	-		ภาพแปลนบ้านชั้น ที่1
h_plan2	VARCHAR(100)	-		ภาพแปลนบ้านชั้น ที่2
h_plan3	VARCHAR(100)	-		ภาพแปลนบ้านชั้น ที่3
h_image	VARCHAR(100)	-		ภาพแบบบ้าน
h_detail	VARCHAR(1000)	-		รายละเอียดบ้าน

ตารางที่ 4.8 รายละเอียดฐานข้อมูลของตาราง Townhouse_Model

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
model_id	INTEGER	Primary Key, Foreign Key	Model	รหัสแบบ อสังหาริมทรัพย์
th_name	VARCHAR(30)	-		ชื่อแบบทาวน์เฮาส์

ตารางที่ 4.8 รายละเอียดฐานข้อมูลของตาราง Townhouse_Model (ต่อ)

th_area	VARCHAR(30)	-		พื้นที่ ทาวเฮาส์
th_floor	INTEGER	-		จำนวนชั้น
th_bedroom	INTEGER	-		จำนวนห้องนอน
th_bathroom	INTEGER	-		จำนวนห้องน้ำ
th_parking	INTEGER	-		จำนวนที่จอดรถ
th_plan1	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 1
th_plan2	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 2
th_plan3	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 3
th_image	VARCHAR(100)	-		ภาพแบบทาวเฮาส์
th_detail	VARCHAR(1000)	-		รายละเอียดทาวเฮาส์

ตารางที่ 4.9 รายละเอียดฐานข้อมูลของตาราง project_status

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
spro_id	INTEGER	Primary Key		รหัสสถานะโครงการ
spro_name	VARCHAR(30)	-		ชื่อสถานะโครงการ

4.2 ฐานข้อมูลของเว็บกลาง

ฐานข้อมูลของเว็บกลางหรือเว็บท่า ใช้ SQLserver 2000 เป็นฐานข้อมูล ประกอบด้วย 3 ตารางดังรูปที่ 4.2 ดังนี้

ตารางที่ 4.8 รายละเอียดฐานข้อมูลของตาราง Townhouse_Model (ต่อ)

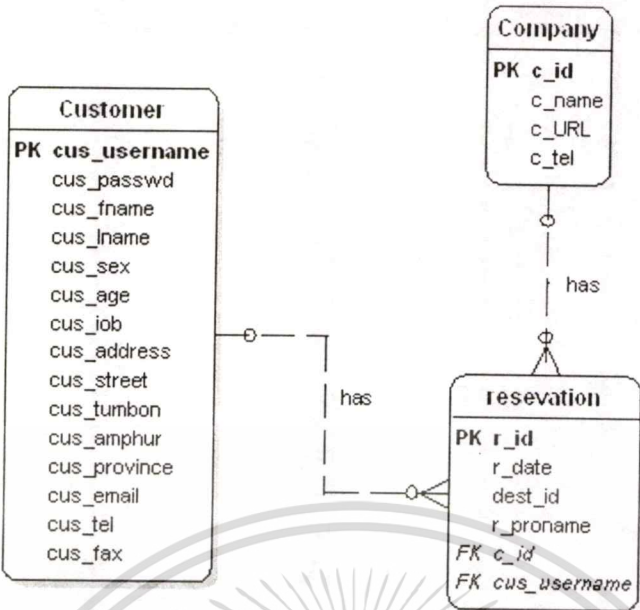
th_area	VARCHAR(30)	-		พื้นที่ ทาวเฮาส์
th_floor	INTEGER	-		จำนวนชั้น
th_bedroom	INTEGER	-		จำนวนห้องนอน
th_bathroom	INTEGER	-		จำนวนห้องน้ำ
th_parking	INTEGER	-		จำนวนที่จอดรถ
th_plan1	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 1
th_plan2	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 2
th_plan3	VARCHAR(100)	-		ภาพแปลนทาวเฮาส์ชั้นที่ 3
th_image	VARCHAR(100)	-		ภาพแบบทาวเฮาส์
th_detail	VARCHAR(1000)	-		รายละเอียดทาวเฮาส์

ตารางที่ 4.9 รายละเอียดฐานข้อมูลของตาราง project_status

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
spro_id	INTEGER	Primary Key		รหัสสถานะโครงการ
spro_name	VARCHAR(30)	-		ชื่อสถานะโครงการ

4.2 ฐานข้อมูลของเว็บกลาง

ฐานข้อมูลของเว็บกลางหรือเว็บท่า ใช้ SQLserver 2000 เป็นฐานข้อมูล ประกอบด้วย 3 ตารางดังรูปที่ 4.2 ดังนี้



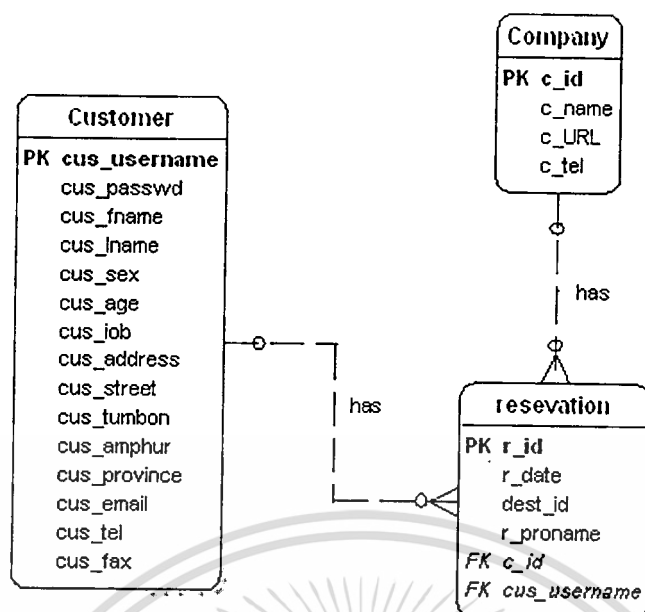
รูปที่ 4.2 ER Diagram ของเว็บกลาง

จากรูปที่ 4.2 มีรายละเอียดดังนี้

- 1. Customer เป็นตารางที่ใช้เก็บข้อมูลลูกค้า โดยรายละเอียดแสดงดังตารางที่ 4.10
- 2. Company เป็นตารางที่ใช้เก็บข้อมูลของบริษัท ที่เปิดบริการ โดยรายละเอียดแสดงดังตารางที่ 4.11
- 3. Reservation เป็นตารางที่ใช้เก็บข้อมูลการจองของลูกค้า โดยรายละเอียดแสดงดังตารางที่ 4.12

ตารางที่ 4.10 รายละเอียดฐานข้อมูลของตาราง Customer

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
cus_username	VARCHAR(30)	Primary Key		ชื่อในการล็อกอิน
cus_passwd	VARCHAR(30)	-		รหัสผ่าน
cus_fname	VARCHAR(30)	-		ชื่อลูกค้า
cus_lname	VARCHAR(30)	-		นามสกุลลูกค้า
cus_sex	VARCHAR(10)	-		เพศ
cus_age	VARCHAR(30)	-		อายุ
cus_job	VARCHAR(30)	-		อาชีพ
cus_address	VARCHAR(30)	-		ที่อยู่
cus_street	VARCHAR(30)	-		ถนน
cus_tumbon	VARCHAR(30)	-		ตำบล
cus_amphur	VARCHAR(30)	-		อำเภอ



รูปที่ 4.2 ER Diagram ของเว็บกลาง

จากรูปที่ 4.2 มีรายละเอียดดังนี้

1. Customer เป็นตารางที่ใช้เก็บข้อมูลลูกค้า โดยรายละเอียดแสดงดังตารางที่ 4.10
2. Company เป็นตารางที่ใช้เก็บข้อมูลของบริษัท ที่เปิดบริการ โดยรายละเอียดแสดงดังตารางที่ 4.11
3. Reservation เป็นตารางที่ใช้เก็บข้อมูลการจองของลูกค้า โดยรายละเอียดแสดงดังตารางที่ 4.12

ตารางที่ 4.10 รายละเอียดฐานข้อมูลของตาราง Customer

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
cus_username	VARCHAR(30)	Primary Key		ชื่อในการล็อกอิน
cus_passwd	VARCHAR(30)	-		รหัสผ่าน
cus_fname	VARCHAR(30)	-		ชื่อลูกค้า
cus_lname	VARCHAR(30)	-		นามสกุลลูกค้า
cus_sex	VARCHAR(10)	-		เพศ
cus_age	VARCHAR(30)	-		อายุ
cus_job	VARCHAR(30)	-		อาชีพ
cus_address	VARCHAR(30)	-		ที่อยู่
cus_street	VARCHAR(30)	-		ถนน
cus_tumbon	VARCHAR(30)	-		ตำบล
cus_amphur	VARCHAR(30)	-		อำเภอ

ตารางที่ 4.10 รายละเอียดฐานข้อมูลของตาราง Customer (ต่อ)

cus_province	VARCHAR(30)	-		จังหวัด
cus_email	VARCHAR(30)	-		อีเมล
cus_tel	VARCHAR(30)	-		เบอร์โทรศัพท์
cus_fax	VARCHAR(30)	-		เบอร์แฟกซ์

ตารางที่ 4.11 รายละเอียดฐานข้อมูลของตาราง Company

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
c_id	INTEGER	Primary Key		รหัสบริษัท
c_name	VARCHAR(30)	-		ชื่อบริษัท
c_URL	VARCHAR(30)	-		URL ของบริษัท
c_tel	VARCHAR(30)	-		เบอร์โทรบริษัท

ตารางที่ 4.12 รายละเอียดฐานข้อมูลของตาราง Reservation

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
r_id	INTEGER	Primary Key		รหัสการจอง
r_date	DATE	-		วันที่จอง
dest_id	INTEGER	-		รหัสโครงการ
r_proname	VARCHAR(30)	-		ชื่อโครงการ
cus_id	INTEGER	Foreign Key	Customer	รหัสลูกค้า
c_id	INTEGER	Foreign Key	Company	รหัสบริษัท

ตารางที่ 4.10 รายละเอียดฐานข้อมูลของตาราง Customer (ต่อ)

cus_province	VARCHAR(30)	-		จังหวัด
cus_email	VARCHAR(30)	-		อีเมล
cus_tel	VARCHAR(30)	-		เบอร์โทรศัพท์
cus_fax	VARCHAR(30)	-		เบอร์แฟกซ์

ตารางที่ 4.11 รายละเอียดฐานข้อมูลของตาราง Company

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
c_id	INTEGER	Primary Key		รหัสบริษัท
c_name	VARCHAR(30)	-		ชื่อบริษัท
c_URL	VARCHAR(30)	-		URL ของบริษัท
c_tel	VARCHAR(30)	-		เบอร์โทรบริษัท

ตารางที่ 4.12 รายละเอียดฐานข้อมูลของตาราง Reservation

ชื่อแอตทริบิวต์	ประเภท	ชนิดคีย์	ตารางอ้างอิง	รายละเอียด
r_id	INTEGER	Primary Key		รหัสการจอง
r_date	DATE	-		วันที่จอง
dest_id	INTEGER	-		รหัสโครงการ
r_proname	VARCHAR(30)	-		ชื่อโครงการ
cus_id	INTEGER	Foreign Key	Customer	รหัสลูกค้า
c_id	INTEGER	Foreign Key	Company	รหัสบริษัท

บทที่ 5

การพัฒนาระบบ

5.1 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ

การพัฒนาระบบในโครงการนี้ได้ใช้เครื่องมือ และภาษาในการพัฒนา ดังนี้

5.1.1 ฮาร์ดแวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนา และทดสอบระบบ โดยมีคุณสมบัติดังนี้

- CPU: Pentium 4 2.4 GHz
- RAM: 512 MB
- Hard Disk: 80 GB
- Network Interface: 10BASE-TX/10BASE-T

5.1.2 ซอฟต์แวร์

ซอฟต์แวร์ที่ใช้ในการพัฒนา และทดสอบระบบ มีดังนี้

- Operation System: Microsoft Windows XP Professional Service Pack 2
- Programming Language: J2SE 1.5.0_06
- Apache Tomcat 6.0.6
- Apache Axis 1.4
- RDBMS: MySQL 5, Microsoft SQLserver 2000 Developer Edition
- Web Browser: Internet Explorer 6.0

5.1.3 เครื่องมือ

เครื่องมือที่ใช้ในการพัฒนาระบบ และทดสอบระบบ มีดังนี้

- Java IDE: Eclipse 3.2
- Web Development Tool: Microsoft Studio 2005 Professional Edition
- Database Design Tool: design v4
- UML Tool: umlet 7.1
- MySQL Database Management Tool: HeidiSQL 3.0 RC4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การพัฒนาระบบ

5.1 เครื่องมือและภาษาที่ใช้ในการพัฒนาระบบ

การพัฒนาระบบในโครงการนี้ได้ใช้เครื่องมือ และภาษาในการพัฒนา ดังนี้

5.1.1 ฮาร์ดแวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนา และทดสอบระบบ โดยมีคุณสมบัติดังนี้

- CPU: Pentium 4 2.4 GHz
- RAM: 512 MB
- Hard Disk: 80 GB
- Network Interface: 10BASE-TX/10BASE-T

5.1.2 ซอฟต์แวร์

ซอฟต์แวร์ที่ใช้ในการพัฒนา และทดสอบระบบ มีดังนี้

- Operation System: Microsoft Windows XP Professional Service Pack 2
- Programming Language: J2SE 1.5.0_06
- Apache Tomcat 6.0.6
- Apache Axis 1.4
- RDBMS: MySQL 5, Microsoft SQLserver 2000 Developer Edition
- Web Browser: Internet Explorer 6.0

5.1.3 เครื่องมือ

เครื่องมือที่ใช้ในการพัฒนาระบบ และทดสอบระบบ มีดังนี้

- Java IDE: Eclipse 3.2
- Web Development Tool: Microsoft Studio 2005 Professional Edition
- Database Design Tool: design v4
- UML Tool: umlet 7.1
- MySQL Database Management Tool: HeidiSQL 3.0 RC4

5.2 รายละเอียดการทำงานของระบบ

โครงการนี้จะมีระบบที่ติดต่อกับผู้ใช้งานอยู่ 3 ระบบคือ เว็บกลางซึ่งเป็นเว็บแอปพลิเคชันที่ไปเรียกบริการ และระบบเว็บเซอร์วิสของเจ้าของโครงการรายที่ 1 และ 2

หน้าจอของเว็บกลางจะประกอบไปด้วยส่วนประกอบหลักๆ ส่วน ดังนี้

1. หน้าจอการสมัครสมาชิกของเว็บกลาง แสดงดังรูปที่ 5.1
2. หน้าจอล็อกอินเข้าสู่ระบบ แสดงดังรูปที่ 5.2
3. หน้าจอแสดงข้อมูลการแจ้งความจำนงค์ของลูกค้าที่เข้าสู่ระบบ แสดงดังรูปที่ 5.3
4. หน้าจอการแก้ไขข้อมูลสมาชิก แสดงดังรูปที่ 5.4
5. หน้าจอการค้นหาตามประเภทของสังฆาริมทรัพย์ แสดงดังรูปที่ 5.5
6. หน้าจอการค้นหาข้อมูลโดยการกำหนดเงื่อนไข แสดงดังรูปที่ 5.6
7. หน้าจอแสดงรายละเอียดของโครงการที่ตรงตามเงื่อนไข แสดงดังรูปที่ 5.7 ซึ่งระบบจะทำการร้องขอไปยังผู้ให้บริการเพื่อนำข้อมูลของโครงการที่ตรงตามเงื่อนไขมาแสดงต่อไป
8. หน้าจอรายละเอียดของแต่ละโครงการ แสดงดังรูปที่ 5.8
9. หน้าจอแผนที่โครงการ แสดงดังรูปที่ 5.9 เป็นรูปแผนที่ที่ได้มาจากระบบของเจ้าของโครงการ
10. หน้าจอแผนที่โครงการสามมิติ แสดงดังรูปที่ 5.10 เป็นแผนที่ที่ได้มาจาก Google map APIs
11. หน้าจอรายละเอียดแบบอสังหาริมทรัพย์ที่มีในโครงการ แสดงดังรูปที่ 5.11
12. หน้าจอการจองโครงการ แสดงดังรูปที่ 5.12 โดยที่เมื่อผู้ใช้ทำการจอง ระบบจะส่งข้อมูลการจองไปยังผู้ให้บริการที่เป็นเจ้าของโครงการและจะทำการบันทึกรายการจองดังกล่าวไว้ที่ระบบเว็บกลางด้วย

5.2 รายละเอียดการทำงานของระบบ

โครงการนี้จะมีระบบที่ติดต่อกับผู้ใช้งานอยู่ 3 ระบบคือ เว็บกลางซึ่งเป็นเว็บแอปพลิเคชันที่ไปเรียกบริการ และระบบเว็บเซอร์วิสของเจ้าของโครงการรายที่ 1 และ 2

หน้าจอของเว็บกลางจะประกอบไปด้วยส่วนประกอบหลักๆ ส่วน ดังนี้

1. หน้าจอการสมัครสมาชิกของเว็บกลาง แสดงดังรูปที่ 5.1
2. หน้าจอล็อกอินเข้าสู่ระบบ แสดงดังรูปที่ 5.2
3. หน้าจอแสดงข้อมูลการแจ้งความจำเป็นของลูกค้ายที่เข้าสู่ระบบ แสดงดังรูปที่ 5.3
4. หน้าจอการแก้ไขข้อมูลสมาชิก แสดงดังรูปที่ 5.4
5. หน้าจอการค้นหาตามประเภทของอสังหาริมทรัพย์ แสดงดังรูปที่ 5.5
6. หน้าจอการค้นหาข้อมูลโดยการกำหนดเงื่อนไข แสดงดังรูปที่ 5.6
7. หน้าจอแสดงรายละเอียดของโครงการที่ตรงตามเงื่อนไข แสดงดังรูปที่ 5.7 ซึ่งระบบจะทำการร้องขอไปยังผู้ให้บริการเพื่อนำข้อมูลของโครงการที่ตรงตามเงื่อนไขมาแสดงต่อไป
8. หน้าจอรายละเอียดของแต่ละโครงการ แสดงดังรูปที่ 5.8
9. หน้าจอแผนที่โครงการ แสดงดังรูปที่ 5.9 เป็นรูปแผนที่ที่ได้มาจากระบบของเจ้าของโครงการ
10. หน้าจอแผนที่โครงการสามมิติ แสดงดังรูปที่ 5.10 เป็นแผนที่ที่ได้มาจาก Google map APIs
11. หน้าจอรายละเอียดแบบอสังหาริมทรัพย์ที่มีในโครงการ แสดงดังรูปที่ 5.11
12. หน้าจอการจองโครงการ แสดงดังรูปที่ 5.12 โดยที่เมื่อผู้ใช้ทำการจอง ระบบจะส่งข้อมูลการจองไปยังผู้ให้บริการที่เป็นเจ้าของโครงการและจะทำการบันทึกรายการจองดังกล่าวไว้ที่ระบบเว็บกลางด้วย

- [Login](#)
- [Register](#)
- [Main search](#)

ลงทะเบียนสมาชิกใหม่

ชื่อ	
นามสกุล	
username	
password	
re-password	
เพศ	☐ ชาย ☐ หญิง
อายุ	
อาชีพ	
ที่อยู่	
ถนน	
ที่อยู่/แขวง	
อำเภอ/เขต	
จังหวัด	
e-mail	
โทรศัพท์	
fax	

ลงทะเบียน

รูปที่ 5.1 หน้าจอการสมัครสมาชิกของเว็บกลาง



- [Login](#)
- [Register](#)
- [Main search](#)

Log In

User Name:

Password:

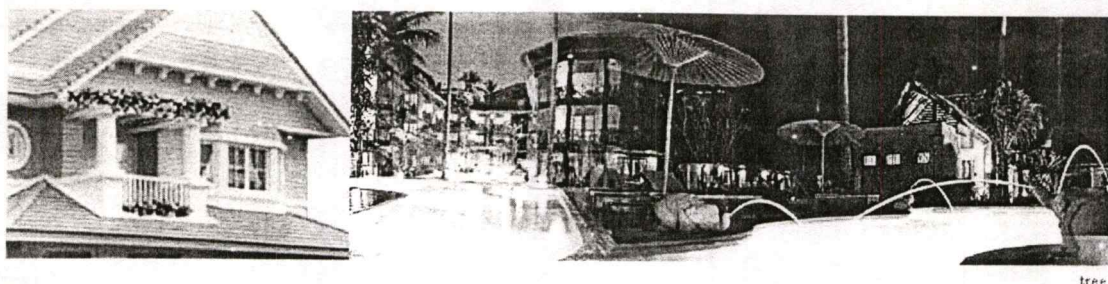
☐ Remember me next time.

Log In

สมัครสมาชิกใหม่

รูปที่ 5.2 หน้าจอล็อกอินเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- o [Login](#)
- o [Register](#)
- o [Main search](#)

สวัสดีครับ คุณ tree
คุณมีรายการแจ้งความจำนงค์นี้

[view your interest](#)
[Edit profile](#)
[Logout](#)

รหัสการจอง	วันที่จอง(mm/dd/yy)	รหัสโครงการ	ชื่อโครงการ	เจ้าของโครงการ
1	30/4/2550 0:00:00	1	มัลเทศนา อ่อนนุช-วงแหวน	comfort property
2	30/4/2550 0:00:00	2	การ์เด็น ลากูนา	happy property
4	1/5/2550 0:00:00	6	บางอ้อ ริเวอร์วิว	comfort property
14	3/5/2550 0:00:00	6	บางอ้อ ริเวอร์วิว	comfort property
18	5/4/3093 0:00:00	1	เกษดานคร วงแหวน อ่อนนุช	happy property

รูปที่ 5.3 ข้อมูลการแจ้งความจำนงค์ของลูกค้าเมื่อเข้าสู่ระบบ



- o [Login](#)
- o [Register](#)
- o [Main search](#)

แก้ไขข้อมูลสมาชิก

[view your interest](#)
[Edit profile](#)
[Logout](#)

ชื่อ	มนตรี
นามสกุล	ปาวชัย
username	tree
เพศ	☒ ชาย ☐ หญิง
อายุ	
อาชีพ	
ที่อยู่	13/18 หมู่ 3
ถนน	ฉลองกรุง
ที่อยู่/แขวง	ลาดกระบัง
อำเภอ/เขต	ลาดกระบัง
จังหวัด	กรุงเทพฯ
e-mail	hhh_077@hotmail.com
โทรศัพท์	084-3336783
fax	02-9456767

รูปที่ 5.4 หน้าจอการแก้ไขข้อมูลสมาชิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- [Login](#)
- [Register](#)
- [Main search](#)

ค้นหาตามประเภทของสิ่งหาหิมทรัพย์



ค้นหาบ้านเดี่ยว

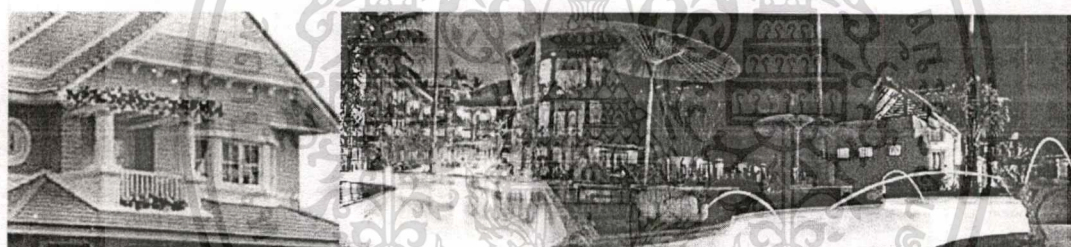


ค้นหาทาวน์เฮาส์



ค้นหาคอนโด

รูปที่ 5.5 หน้าจอการค้นหาตามประเภทของอสังหาริมทรัพย์



- [Login](#)
- [Register](#)
- [Main search](#)

ค้นหาบ้าน

เจ้าของโครงการ

ทั้งหมด

[view your interest](#)
[Edit profile](#)
[Logout](#)

จังหวัด

กรุงเทพฯ

ทำเล

คลองเตย

งบประมาณ

C ต่ำกว่า 1 ล้านบาท	C 1.1-2.1 ล้านบาท
C 2.1-3.1 ล้านบาท	C 3.1-5.1 ล้านบาท
C 5.1-7.1 ล้านบาท	C 7.1-10.1 ล้านบาท
C 10.1-15.1 ล้านบาท	C 15.1-20.1 ล้านบาท
C 20.1 ล้านบาทขึ้นไป	

ค้นหา

รูปที่ 5.6 หน้าจอการค้นหาข้อมูลโดยการกำหนดเงื่อนไข
 เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- o [Login](#)
- o [Register](#)
- o [Main search](#)

[Edit profile](#)
[Logout](#)

ชื่อโครงการ

มักเจนา อ่อนนุช-วงแหวน

เจ้าของโครงการ

comfort property



ราคาเริ่มต้น

4.44 ล้านบาท

ขนาดพื้นที่โครงการ

168.0 ไร่

สถานะของโครงการ

สร้างเสร็จ

ชื่อโครงการ

กฤษดานคร วงแหวน อ่อนนุช

เจ้าของโครงการ

happy property



ราคาเริ่มต้น

3.89 ล้านบาท

รูปที่ 5.7 หน้าจอแสดงรายละเอียดของโครงการที่ตรงตามเงื่อนไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



tree

- [Login](#)
- [Register](#)
- [Main search](#)

[view your interest](#)
[Edit profile](#)
[Logout](#)

กฤษดานคร วงแหวน อ่อนนุช



ราคาเริ่มต้นที่ 3.89 ล้านบาท

พื้นที่โครงการ

0.0 ไร่

สถานที่แวดล้อม

- สนามบินสุวรรณภูมิ
- ถนนวงแหวนรอบนอก(บางนา-บางปะอิน)

วัตถุประสงค์ของโครงการ

โครงการใหม่ล่าสุดจากกฤษดานคร ที่ซึ่งพร้อมให้ท่านและครอบครัวสามารถสัมผัสกับบรรยากาศเพื่อการพักผ่อน และผ่อนคลายอย่างสมบูรณ์แบบสไตล์ Oriental Spa อันสงบบรมรื่น พร้อมบ้านดีไซน์ใหม่ Thai Modern Contemporary ที่คุ้มค่าด้วย Spec และ Spec ที่มากกว่า



ดูภาพขยาย
ดัดภาพ 3 มิติ

รายละเอียดของโครงการ

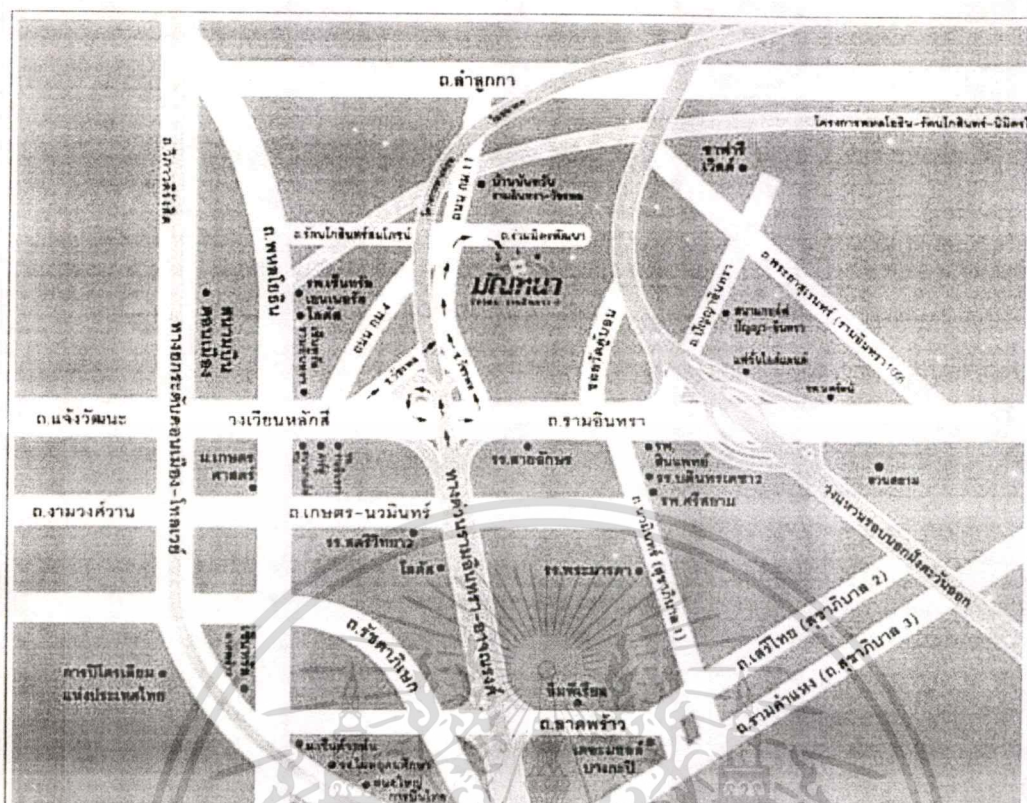
การคมนาคมที่สะดวก สามารถเดินทางเข้าเมืองได้อย่างรวดเร็ว โดยถนนมอเตอร์เวย์ และถนนวงแหวนรอบนอก (บางนา-บางปะอิน) ใกล้สนามบินสุวรรณภูมิ เพียง 5 นาที เป็นโครงการขนาดเล็ก ที่มีความสงบและความเป็นส่วนตัว รั้วรอบกำแพงหนาแน่น มีสิ่งอำนวยความสะดวกครบครัน อาทิ สวนสาธารณะ สโมสร สระว่ายน้ำ

โฉมโครงการ

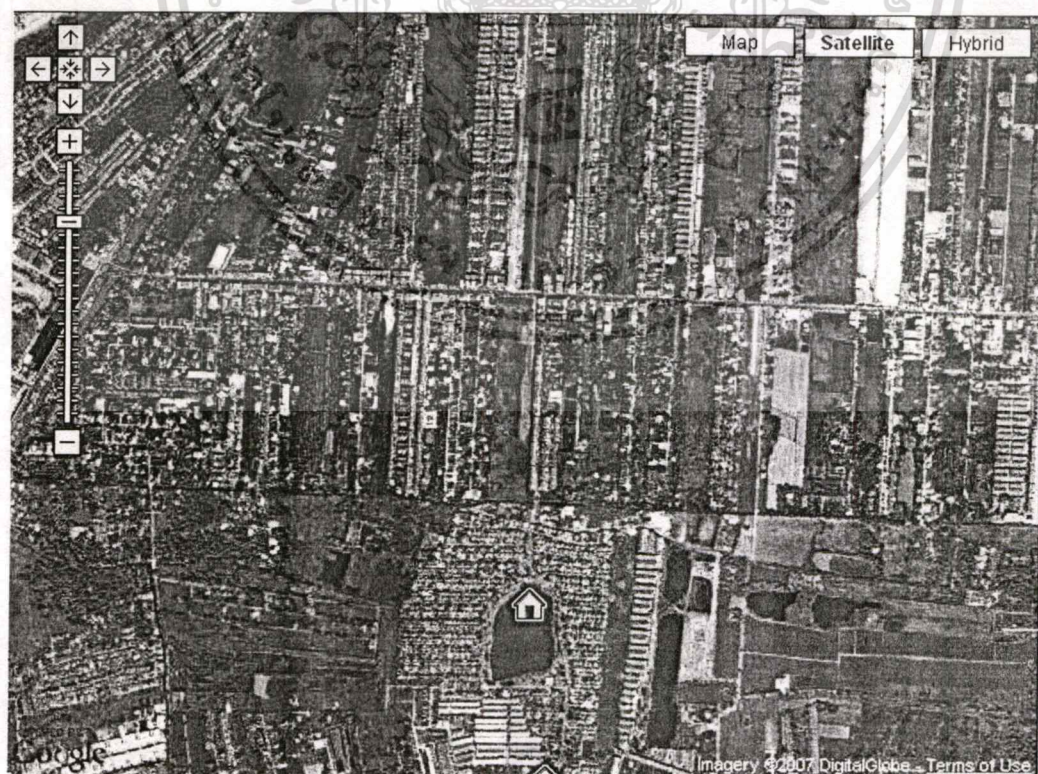
คลิก Select เพื่อดูรายละเอียด

แบบบ้าน	ราคา(ล้านบาท)	ดาวน์	เงินจอง	ทำสัญญา	เงินโอน(ล้านบาท)	ผ่อนเริ่มต้นต่อเดือน	ดอกเบี้ย (%)	ปี
1	4.25	0.0	0.0	0.0	0.0	0.0	0.0	0 Select
2	4.97	0.0	0.0	0.0	0.0	0.0	0.0	0 Select
3	3.89	0.0	0.0	0.0	0.0	0.0	0.0	0 Select

เอกสารนี้เป็นเอกสารที่ส่งในรูปแบบที่ 5.8 หน้าจ้อรายละเอียดของแต่ละโครงการภายใต้เงื่อนไขการใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



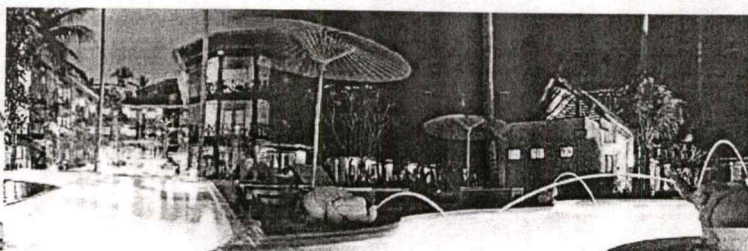
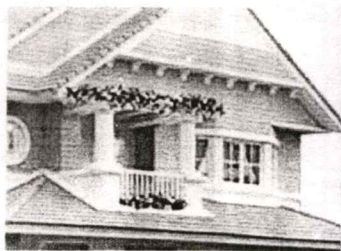
รูปที่ 5.9 หน้าจอแผนที่โครงการ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้อ้างอิงเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 5.10 หน้าจอแผนที่โครงการสามัคคี

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



tree

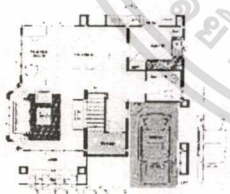
- [Login](#)
- [Register](#)
- [Main search](#)

[view your
interest](#)
[Edit profile](#)
[Logout](#)

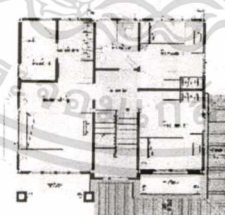
แบบบ้าน จุฬาพิมล
รหัสแบบบ้าน 2



พื้นที่ใช้สอย 156.5 ตร.ม.
ขนาดที่ดิน 60.9 ตร.ว.
จำนวนชั้น 2 ชั้น
จำนวนห้องนอน 3 ห้อง
จำนวนห้องน้ำ 3 ห้อง
จำนวนที่จอดรถ 2 คัน
รถ
ลักษณะเด่น 1 ห้องรับแขก 1 ห้องรับประทานอาหาร 1 ห้องครัว 1 ห้องเก็บของ



plan ชั้นที่ 1



plan ชั้นที่ 2

รูปที่ 5.11 หน้าจอรายละเอียดแบบก่อสร้างที่มียกในโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



tree

- o [Login](#)
- o [Register](#)
- o [Main search](#)

[view your interest](#)

[Edit profile](#)

[Logout](#)

แจ้งความสนใจโครงการ

ชื่อ	มนตรี
นามสกุล	บุราชัย
เพศ	male
อาชีพ	
เบอร์โทรศัพท์	084-3336783
e-mail	hhh_077@hotmail.com
ชื่อโครงการ	กฤษดากร วงแหวน อ่อนนุช
บริษัทเจ้าของโครงการ	happy property

รายละเอียดเพิ่มเติม

แจ้งความจำน

รูปที่ 5.12 หน้าจอการจองโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทสรุป

6.1 สรุปผลการออกแบบและพัฒนาระบบงาน

ในโครงการนี้ได้ดำเนินการศึกษาข้อมูล ทฤษฎี และมาตรฐานเทคโนโลยีที่เกี่ยวข้องกับเว็บเซอร์วิส โดยนำความรู้ที่ได้มาออกแบบและพัฒนาระบบการค้นหาและพาณิชย์อิเล็กทรอนิกส์รูปแบบใหม่ ทั้งเว็บแอปพลิเคชัน และเว็บเซอร์วิส โดยการยกตัวอย่างการนำจาวาเว็บเซอร์วิสและค็อดเน็ตเว็บเซอร์วิสการพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังหาริมทรัพย์ เพื่อทดสอบว่าเทคโนโลยีเว็บเซอร์วิสสามารถทำงานข้ามแพลตฟอร์มที่ต่างกันได้และช่วยให้ระบบการพาณิชย์อิเล็กทรอนิกส์มีประสิทธิภาพยิ่งขึ้น

ในช่วงการวิเคราะห์จนถึงการออกแบบ ได้ทำการศึกษาทฤษฎีสำหรับการออกแบบและพัฒนา ระบบเพิ่มเติมคือ Unified Modeling Language (UML) จากนั้นจึงทำการออกแบบ และพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังหาริมทรัพย์ ซึ่งในระหว่างการพัฒนา ระบบ ได้มีการทดสอบ ปรับปรุงโครงสร้าง และฟังก์ชันการทำงานของระบบ จนสามารถใช้งานได้ตรงตามความต้องการ

6.2 ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ

ประโยชน์ที่ได้รับจากการพัฒนาระบบโครงการนี้สามารถสรุปได้ดังนี้

1. ได้ศึกษาเทคโนโลยีเว็บเซอร์วิสทั้งค็อดเน็ตและจาวาเว็บเซอร์วิส แล้วนำมาประยุกต์ใช้ในการพัฒนาระบบการพาณิชย์อิเล็กทรอนิกส์
2. ได้ความรู้ และทักษะในการออกแบบ และพัฒนาระบบงานด้วย UML โดยที่สามารถนำไปประยุกต์ใช้ในการออกแบบระบบงานอื่นๆได้
3. ได้ศึกษาการใช้งาน Google map APIs มาประยุกต์ใช้งานในโครงการเว็บเซอร์วิสการพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังหาริมทรัพย์
4. ได้ระบบการค้นหาข้อมูลรูปแบบใหม่ที่มีประสิทธิภาพมากขึ้น นั่นคือสามารถทำให้การติดต่อสื่อสารระหว่างองค์กรทำได้ง่ายขึ้น โดยที่ไม่ต้องคำนึงถึงเรื่องของแพลตฟอร์มที่ทำงานอยู่
5. ได้ศึกษาแนวทางในการพัฒนาเว็บเซอร์วิสการพัฒนาเว็บเซอร์วิสเพื่อการเลือกซื้อสังหาริมทรัพย์รูปแบบใหม่ เพื่อนำไปประยุกต์ใช้ในการพัฒนาระบบการพาณิชย์อิเล็กทรอนิกส์อื่นๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ข้อจำกัดของระบบที่ออกแบบและพัฒนาขึ้น

ข้อจำกัดของระบบที่ออกแบบและพัฒนาขึ้นในโครงการนี้สามารถสรุปได้ดังนี้

1. Google map APIs เป็นGIS แบบ3D ที่เพิ่งเปิดตัวได้ไม่นาน จึงทำให้การทำงานบางส่วนยังไม่สมบูรณ์ เช่นไม่สามารถรองรับ ไอคอนกราฟิก ในบางรูปได้จึงทำให้การหาตำแหน่งโครงการต้องใช้ความพยายามในการค้นหามากขึ้น และเนื่องจากเป็นแผนที่แบบสามมิติที่ทำงานบนเว็บจึงทำให้การทำงานช้า
2. ในการจำลองเพื่อสร้างเว็บ server ของ apache Tomcat ให้ทำงานสองตัวพร้อมกันไม่สามารถทำได้ด้วยการเปลี่ยนแค่หมายเลขพอร์ต จะต้องใช้ การ์ดแลนสองตัวในการทำ จึงจำลองโดยการ สร้างเซิร์ฟเวอร์สองเซิร์ฟเวอร์บนserver เดียวกันแต่ทั้งสองเซิร์ฟเวอร์จะไม่เกี่ยวข้องกัน

6.4 ปัญหาและอุปสรรคระหว่างการออกแบบและพัฒนาระบบงาน

ปัญหาและอุปสรรคระหว่างการพัฒนาระบบงานในโครงการนี้สามารถสรุปได้ดังนี้

1. เนื่องจากเทคโนโลยีที่ใช้ในโครงการนี้เป็นเทคโนโลยีที่ใหม่ และต้องใช้เวลาในการทำ ความเข้าใจและการพัฒนา โดยเฉพาะในส่วนของจาวาเว็บเซิร์ฟเวอร์ ดังนั้นจึงเสียเวลาในการศึกษาเทคโนโลยีมาก
2. การพัฒนาระบบงานจะทำการพัฒนาในเครื่องเดียวทั้งๆที่ต้องมีการติดต่อกับผู้ให้บริการมากกว่าหนึ่ง จึงทำให้ต้องมีการจำลองในส่วนที่เหลือ
3. ในส่วนของ Google map APIs จะทำงานได้ช้าเพราะเป็นแผนที่สามมิติที่ทำงานบนเว็บ

6.5 ข้อเสนอแนะและแนวทางในการพัฒนาระบบเพิ่มเติม

ระบบงานที่พัฒนาขึ้นในโครงการนี้ ถึงแม้จะเป็นระบบรูปแบบใหม่ที่มีประสิทธิภาพมากขึ้นกว่าเดิม แต่ก็ยังคงมีจุดที่ต้องปรับปรุง ดังนี้

1. ควรเพิ่มความสามารถในเรื่องของการค้นหาสถาบันการเงิน ที่มีความสำคัญในเรื่องของ กู้ยืมเงิน รวมทั้งอัตราดอกเบี้ยต่างๆ
2. ควรมีการคำนวณเงินที่จะต้องจ่ายในแต่ละงวดถ้าหากตัดสินใจเลือกซื้อ โครงการดังกล่าว
3. ควรมีการเก็บสถิติการเข้าชม สถิติการจอง โครงการที่ได้รับความนิยมและลำดับความนิยมของบริษัทที่มีการสั่งจอง เพื่อเป็นข้อมูลในการตัดสินใจให้แก่ผู้ใช้ได้อีกทางหนึ่งด้วย

บรรณานุกรม

นรินทร์ โอพารกิจอนันต์. ม.ป.ป.. **J2EE&XML จาวาระดับองค์กร**. กรุงเทพฯ: เติชูกิ ดอทเน็ต.

ชิน ภู่วรรณ. 2550. **.NET กำลังมา**. [Online] Available:

http://micro.se-ed.com/content/mc207/MC207_143.asp.

สราวุธ อ้อยศรีสกุล. 2544. **เริ่มคิด-เริ่มสร้าง-เริ่มใช้ XML**. กรุงเทพฯ: วิตดีกรุ๊ป.

สวัสดิษฐ์ เกรียงไกรเพชร. 2549. **การใช้ Google Maps API เพื่อสร้าง Google Map Mashup**

อย่างง่าย. [Online] Available:

<http://lecturer.cng.chula.ac.th/fsvskk/gglmap2/get-start-gmap-apiv2.php>.

Ed Ort. 2005. **Service-Oriented Architecture and Web Services: Concepts, Technologies and Tools**. [Online]. Available:

<http://java.sun.com/developer/technicalArticles/WebServices/soa2/index.html>.

Kregeri, H. 2001. **Web Services Conceptual Architecture (WSCA 1.0)**. [Online]. Available:

<http://www-306.ibm.com/software/solutions/webservices/pdf/WSCA.pdf>.

Nagappan, R. et al. 2003. **Developing Java Web Services**. Singapore: Wiley Publishing.

OASIS Open. 2002. **UDDI Version 2.04 API Specification**. [Online]. Available:

<http://uddi.org/pubs/ProgrammersAPI-V2.04-Published-20020719.htm>.

ภาคผนวก ก

ขั้นตอนการสมัครขอใช้บริการ Google Maps API



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการสมัครขอใช้บริการ Google Maps API

Google Maps API ช่วยให้เราสามารถพัฒนาโปรแกรมเพื่อแทรก Google Maps เข้าไปเป็นองค์ประกอบส่วนหนึ่งในเว็บเพจที่ต้องการได้โดยเขียนเป็นรหัส html และ javascript ในรูปแบบที่ไม่สลับซับซ้อนนักสำหรับงานแผนที่ง่ายๆ Google Maps API มีขีดความสามารถกว้างขวางเน้นในด้านการนำเสนอข้อมูลแผนที่ในลักษณะ Push pin หรือ Place marker ซึ่งสามารถกำหนดให้แสดงข้อมูลประกอบแผนที่เมื่อผู้ใช้คลิกที่ตัว pushpin /marker นั้นๆ บริการเรื่องแผนที่ของ Google นี้เริ่มต้นตั้งแต่กลางปีค.ศ. 2005 เป็นบริการฟรี จัดให้แก่ผู้ใช้ที่ร้องขอโดยคาดหวังที่จะใช้การโฆษณาบนแผนที่เป็นรายได้กลับคืน แต่ในระยะแรกจะยังไม่มีโฆษณาดังกล่าว

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

บทความนี้จะกล่าวถึงการใช้ Google Maps API ทำโปรแกรมประเภทที่เรียกว่า Map mashup อย่างง่ายเพื่อเป็นตัวอย่างให้ผู้สนใจได้ทำขึ้นเพื่อเริ่มต้นทดลองใช้งาน หรือเรียนรู้ทำความเข้าใจในเทคโนโลยีและขีดความสามารถของ Google Maps API

Google Maps mashup

คำว่า “mashup” หรือ “mash-up” เป็นศัพท์เฉพาะในแวดวงของนักพัฒนา application แบบ client-server ประเภทที่ทำเดิมต่อขึ้นเองโดยใช้ software tool และทรัพยากรที่ทางผู้ให้บริการจัดไว้ให้ เป็นคำใหม่ที่ใช้กันหนาหูในปี 2005 มีที่มาจากวงการเพลงเมื่อนักจัดรายการเพลงเกิดความคิดแหวกแนวโดยทดลองเปิดเพลง 2 เพลงพร้อมๆ กัน ทำให้เกิดเสียงเพลงที่ผสมผสานที่แปลก และบางครั้งมีความไพเราะ และเรียกขานผลที่เกิดขึ้นว่า “mashup”

mashup ที่กล่าวถึงกันมากในแวดวงนักพัฒนาแอปพลิเคชันที่ใช้บนอินเทอร์เน็ตมีองค์ประกอบอย่างน้อย 2 ส่วน ส่วนแรกคือ application program interface (API) และทรัพยากรสนับสนุนจากเซิร์ฟเวอร์ของผู้ให้บริการ และส่วนที่ 2 คือโปรแกรมและทรัพยากรของผู้พัฒนาแอปพลิเคชันนั้น (ติดตั้งอยู่ที่เว็บไซต์ของผู้พัฒนา) ตัวอย่างกรณีของ Google Maps mashup ก็จะหมายถึงเว็บเพจที่นักพัฒนาเว็บไซต์สร้างขึ้นให้มีส่วนประกอบที่เป็นแผนที่ที่สามารถทำงานแบบโต้ตอบกับผู้ใช้ได้ โดยมีแผนที่ฐานเป็นส่วนที่ Google จัดไว้ให้แล้ว และมีข้อมูลแผนที่ของตนเองไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซ้อนทับในลักษณะหมุดปัก (ที่สามารถแสดงข้อความซึ่งแฝงอยู่เมื่อผู้ใช้คลิกที่หมุดเหล่านั้น) หรือลักษณะอื่นที่สลับซับซ้อนกว่านั้น

ความรู้ความชำนาญที่ควรมีในผู้ใช้ Google Maps API

ในการจัดทำ Google Maps mashup ระดับต้นจำเป็นต้องใช้ความรู้พื้นฐานในการสร้าง web page ด้วยภาษา html/ xhtml การเขียนโปรแกรม javascript ความรู้เรื่องการแผนที่และภูมิศาสตร์ และความรู้ในด้าน xml technology ในระดับปฏิบัติการที่สูงขึ้นมีความต้องการที่สลับซับซ้อนมากขึ้นตามลำดับ

ในที่นี้ขอก้าวโดยสรุปเกี่ยวกับความรู้พื้นฐานเบื้องต้นที่จำเป็นในงานพัฒนา Google Maps mashup ซึ่งควรมี ดังนี้

1. html/ xhtml
2. javascript
3. ความรู้วิชาภูมิศาสตร์ และการแผนที่

ความรู้เพิ่มเติมที่จำเป็นสำหรับงานพัฒนาขั้นกลาง-สูง

1. XML technology
2. AJAX (javascript + xml)
3. JSON (JavaScript Object Notation)

ความรู้เพิ่มเติมสำหรับงานใหญ่ งานพิเศษ

1. Server side scripting (PHP, etc)
2. Web database software (MySQL, etc)

ความรู้อื่นที่เกี่ยวข้อง มีทั้งที่เป็นเรื่องกราฟิก แผนที่ และ อื่นๆ

1. Web map service (WMS) concept
2. SVG (Scalable Vector Graphic)
3. GML (Geographic Markup Language)
4. GIS, GPS, Cartography
5. etc.

ขั้นตอนหลักในการดำเนินการจัดทำ Google Maps mashup

1. ยื่นคำขอ API Key สำหรับ URL ของเว็บไซต์ที่จะใช้บริการ Google Maps API เว็บไซต์นั้นเป็นที่ที่เราจะนำเว็บเพจที่มี Google Maps เป็นองค์ประกอบไปวางไว้ ดังรูปที่ 1

(URL ที่ยื่นคำขอรับ API Key:- <http://www.google.com/apis/maps/signup.html>)

Google Maps API Terms of Use

Thank you for using the Google Maps API! By using the Google Maps API (the "Service"), you ("You") accept and agree to be bound by the Terms of Service for Google Maps as well as these additional terms and conditions (the "Terms of Use").

1. Service.

1.1 Description of Service. The API consists of Javascript that allows You to display Google map images on your website, subject to the limitations and conditions described below. The API is limited to allowing You to display map images only, and does not provide You with the ability to access the underlying map data, any services provided by Google in connection with its maps service (such as local

☐ I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

รูปที่ 1 หน้าจอการขอ API Key

2. เขียนโปรแกรมสร้างเว็บเพจที่ใช้ Google Maps API ช่วยสร้างแผนที่

(แหล่งเอกสารอ้างอิง:- <http://www.google.com/apis/maps/documentation/reference.html>)

3. Upload งานที่ทำเสร็จแล้วขึ้นสู่เว็บไซต์ และทดลองใช้งาน

เค้าโครงของแฟ้มข้อมูล HTML สำหรับการสร้าง Google Maps Mashup อย่างง่าย

เนื่องจากการแสดงตัวอย่างในบทความนี้ละเว้นส่วนที่มีซ้ำกันเพื่อให้ประหยัดเนื้อที่ และมีความกระชับมากขึ้น ดังนั้น ส่วนที่เป็นเค้าโครงแฟ้มข้อมูลซึ่งต้องใช้ในทุกตัวอย่างจึงเป็นสิ่งที่สำคัญที่ควรกล่าวถึงก่อน จึงขอนำมาแสดงไว้ ดังนี้

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
```

```
<title>My Google Maps Mash-up</title>
```

```
<style type="text/css">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

โดยไม่ได้รับอนุญาต สิ่งนี้จะไม่ให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
behavior:url(#default#VML);
```

```

}

</style>

<script src="http://maps.google.com/maps?file=api&v=2&key=abcdefg..."
type="text/javascript"></script>

</head>

<body onload='onloadFunction()' onunload='unloadGmap()>

<!-- body content here -->

<!-- some javascript here -->

</body>

</html>

```

ข้อความ "v=2" ซึ่งแทรกอยู่ใน "http://maps.google.com/maps?file=api&v=2 ..." เป็นพารามิเตอร์ระบุความต้องการขอใช้ Google Maps API เวอร์ชัน 2 เพื่อทำการประมวลผลในส่วน <!-- body content here --> ควรมี <div id='map' /> เพื่อทำหน้าที่เป็นขอบเขตแสดงผลแผนที่ และ <!-- some javascript here --> เตรียมไว้เป็นที่ของจาวาสคริปต์ซึ่งจะนำมาแทรกภายหลัง

ตัวอย่างพื้นฐานเพื่อการเริ่มต้น

ตัวอย่างทั้งหมดที่ยกมาแสดงในที่นี้จะมีให้เฉพาะส่วนประกอบที่เกี่ยวข้องกับการใช้ Google Maps API เท่านั้นเพื่อความกระชับ ทั้งนี้ยกเว้นตัวอย่างแรกที่ทำหน้าที่ kick start ให้กับผู้ใช้ นอกจากนั้นยังได้ละเว้นส่วนประกอบของโปรแกรมที่ทำหน้าที่สำคัญที่ควรมีในการทำงานจริง เช่น ตรวจสอบความสามารถของเบราว์เซอร์ที่ผู้ใช้ติดตั้งอยู่ว่าทำงานเข้ากันได้หรือไม่ การตรวจสอบป้องกันการทำงานผิดพลาด การทำให้มีคุณลักษณะเป็นมิตรกับผู้ใช้ ฯลฯ

ตัวอย่าง

ตัวอย่างแรกนี้แสดงการสร้างแผนที่อย่างง่ายให้ศูนย์กลางขอบเขตแผนที่อยู่ที่เมือง Palo Alto แห่งรัฐแคลิฟอร์เนียซึ่งมีพิกัดภูมิศาสตร์ ละติจูด / ลองจิจูดเป็น 37.4419, -122.1419 องศาตามลำดับ (สำหรับกรุงเทพฯ ให้ใช้ 13.80608, 100.57983) ซึ่งแสดงดังโค้ดด้านล่าง

```

<!-- This web page displays a 500x300 map centered on Palo Alto,
California: -->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8"/>
<title>Google Maps JavaScript API Example</title>
<script
src="http://maps.google.com/maps?file=api&v=2&key=abcdefg"

```

```

type="text/javascript"></script>
<script type="text/javascript">
//

function load() {
if (GBrowserIsCompatible()) {
var map = new GMap2(document.getElementById("map"));
map.setCenter(new GLatLng(37.4419, -122.1419), 13);
}
}
//]]&gt;
&lt;/script&gt;
&lt;/head&gt;
&lt;body onload="load()" onunload="GUnload()"&gt;
&lt;div id="map" style="width: 500px; height: 300px"&gt;&lt;/div&gt;
&lt;/body&gt;
&lt;/html&gt;
</pre>
</div>
<div data-bbox="140 352 580 377" data-label="Text">
<p>สำหรับผลลัพธ์ที่เกิดขึ้นจากโค้ดข้างต้นเป็นดังรูปที่ 2</p>
</div>
<div data-bbox="64 302 804 696" data-label="Image">
<img alt="A map of the Palo Alto area, California, showing major roads like Middlefield Rd, University Ave, and Highway 82. Landmarks such as Stanford University, Stanford Stadium, and Palo Alto Municipal Golf Course are labeled. The map is overlaid with a large, faint circular watermark featuring a Thai temple and the text 'มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี' (Mahavithayalai King Mongkut's Institute of Technology Srinakharinwirot)."/>
</div>
<div data-bbox="312 697 621 722" data-label="Caption">
<p>รูปที่ 2 แผนที่จาก Google Map APIs</p>
</div>
<div data-bbox="45 917 950 970" data-label="Text">
<p>เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า<br/>ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้</p>
</div>
```