

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์

WEB CONTENT MONITORING SYSTEM WITH MULTI-AGENTS



H006304

โดย

มะลิวัลย์ เลাহกานต์นิยม

MALIWAN LAOHAKANNIYOM

อาจารย์ที่ปรึกษา

ผศ.ดร. ภัทรชัย สลิตโรจน์วงศ์

เลขหมู่.....  
เลขทะเบียน 06304  
วัน,เดือน,ปี: 17 ก.พ. 2554

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

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาอิสระ

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2552

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# WEB CONTENT MONITORING SYSTEM WITH MULTI-AGENTS



**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS OF THE COURSE**

**INDEPENDENT STUDY**

**MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY**

**FACULTY OF INFORMATION TECHNOLOGY**

**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2/ 2009



**COPYRIGHT 2010**

**FACULTY OF INFORMATION TECHNOLOGY**

**KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้ณาเบเซบระเยชนตนาการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ใบรับรองการศึกษาอิสระ (Independent Study)

เรื่อง


## ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ Web Content Monitoring System with Multi-Agents

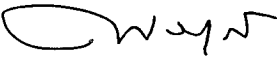
นางมะลิวัลย์ เลาทกานต์นิยม

รหัสประจำตัว 51066402

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด  
รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ  
การศึกษาวិชาการศึกษาอิสระ หลักสูตรวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ)  
ภาคเรียนที่ 2 ปีการศึกษา 2552

  
.....อาจารย์ที่ปรึกษา  
( ผศ.ดร. ภัทรชัย ลลิตโรจน์วงศ์ )

  
.....กรรมการสอบ  
( รศ.ดร. วรพงษ์ กริสุระเดช )

  
.....กรรมการสอบ  
( ผศ.ดร. พรฤดี เนติโสภาคกุล )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์
นักศึกษา	นาง มะลิวัลย์ เลาหกานต์นิยม
รหัสนักศึกษา	51066402
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2552
อาจารย์ที่ปรึกษา	ผศ.ดร. ภัทรชัย ลลิตโรจน์วงศ์

### บทคัดย่อ

รายงานฉบับนี้เสนอผลการศึกษา และพัฒนาระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ ซึ่งมีวัตถุประสงค์เพื่อใช้เป็นเครื่องมือในการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลข่าวสารต่างๆที่ผู้ใช้สนใจ และทำการแจ้งผลการเปลี่ยนแปลงให้ผู้ใช้ทราบผ่านแอปพลิเคชัน โดยระบบจะพัฒนาด้วยหลักการทำงานของมัลติเอเจนต์ในรูปแบบของเอเจนต์ที่ทำงานร่วมกัน นอกจากนี้ระบบจะออกแบบเพื่อช่วยลดปริมาณข้อมูลและการสื่อสารในเครือข่าย โดยในส่วนของเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล จะใช้รูปแบบการทำงานแบบเอเจนต์เคลื่อนที่ เพื่อช่วยเพิ่มประสิทธิภาพเพิ่มช่องทางเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล และประหยัดเวลาในการเข้าถึงข้อมูลที่เปลี่ยนแปลงบนเว็บไซต์ โดยจะพัฒนาระบบบนเจดแพลตฟอร์ม

<b>Title</b>	Web Content Monitoring System with Multi-Agents
<b>Student</b>	Mrs. Maliwan Laohakaniyom
<b>Student ID.</b>	51066402
<b>Degree</b>	Master of Science
<b>Program</b>	Information Science
<b>Major</b>	Information Technology
<b>Academic Year</b>	2009
<b>Advisor</b>	Asst. Prof. Dr. Pattarachai Lalitrojwong

## ABSTRACT

This report describes a result from the study and development of multi-agent web content monitoring system. It is intended to be used as a tool for monitoring the information on the web page that users are interested. The agent notifies the users when the content are changed. This system employs multi-agent and mobile agent technology. This solution improves system performance and reduces the traffic on the network. The system is developed using Java Agent Development Framework (JADE).

# กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. ภัทรชัย ลลิตโรจน์วงศ์ ซึ่งเป็นอาจารย์ผู้ให้คำปรึกษาในโครงการนี้ ข้าพเจ้ารู้สึกทราบบ้างในความอนุเคราะห์จากท่านอาจารย์ และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ภาควิชาวิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า ขอขอบคุณ บริษัท ที แมกซิเนอริ (ประเทศไทย) จำกัด ที่ได้สนับสนุนทุนการศึกษา ตลอดจนเครื่องมือ ข้อมูล และหนังสือต่างๆ ที่ใช้ในการทำโครงการ

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิทยาการสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัยคณะเทคโนโลยีสารสนเทศที่ให้ความช่วยเหลือในเรื่องต่างๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง ทำให้ข้าพเจ้าสามารถทำโครงการฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากโครงการฉบับนี้ ข้าพเจ้าขอบแต่ผู้มีพระคุณทุกท่าน

มะลิวัลย์ เล่าหกานต์นิยม

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ขอบเขตการพัฒนาระบบ.....	2
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.4 ขั้นตอนการพัฒนาระบบ.....	3
บทที่ 2 เทคโนโลยีและงานวิจัยที่เกี่ยวข้อง.....	5
2.1 เอเจนต์.....	5
2.2 มัลติเอเจนต์.....	6
2.3 เจด.....	8
2.4 งานวิจัยที่เกี่ยวข้อง.....	10
บทที่ 3 การวิเคราะห์ระบบงานปัจจุบัน.....	18
3.1 เว็บไซต์ Watch That Page.....	18
3.2 เว็บไซต์ The Web Watcher.....	20
3.3 โปรแกรม WebMon.....	22
3.4 ลักษณะการทำงานของระบบในปัจจุบัน.....	25
3.5 ปัญหาที่พบในระบบปัจจุบัน.....	26
3.6 ความต้องการของระบบงานใหม่.....	26
บทที่ 4 การวิเคราะห์และออกแบบระบบงานใหม่.....	28
4.1 ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บ.....	28

# สารบัญ (ต่อ)

	หน้า
4.2 ยูสเคสไดอะแกรม.....	29
4.3 คลาสไดอะแกรม.....	40
4.3 สเตทชาร์ตไดอะแกรม.....	42
4.4 ซีควเอนซ์ไดอะแกรม.....	47
4.5 การออกแบบการจัดเก็บข้อมูล.....	50
บทที่ 5 การออกแบบและพัฒนาระบบ.....	53
5.1 เครื่องมือและภาษาที่ใช้ในการพัฒนา.....	53
5.2 การจำลองเอเจนต์เซิร์ฟเวอร์.....	54
5.3 การพัฒนาระบบเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์.....	55
บทที่ 6 บทสรุป.....	59
6.1 สรุปโครงการ.....	59
6.2 ปัญหาและอุปสรรคระหว่างการออกแบบและพัฒนาระบบงาน.....	59
6.3 ข้อจำกัดของระบบ.....	59
บรรณานุกรม.....	61
ประวัติผู้เขียน.....	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา V ละต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง

ตารางที่	หน้า
4.1 คำอธิบายแอกเตอร์ของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์.....	30
4.2 คำอธิบายยูสเคสไดอะแกรมของการ Add URL for monitoring.....	30
4.3 คำอธิบายยูสเคสไดอะแกรมของการ Set keyword filtering .....	32
4.4 คำอธิบายยูสเคสไดอะแกรมของการ Set alert channels .....	33
4.5 คำอธิบายยูสเคสไดอะแกรมของการ Set alert message .....	35
4.6 คำอธิบายยูสเคสไดอะแกรมของการ View content change summary report .....	36
4.7 แอกเตอร์ที่เกี่ยวข้องกับ Web Fetcher Agent .....	38
4.8 คำอธิบายยูสเคสไดอะแกรมของการ Check file property .....	38
4.9 คำอธิบายยูสเคสไดอะแกรมของการ Notify Content Change .....	38
4.10 แอกเตอร์ที่เกี่ยวข้องกับ Server Agent .....	39
4.11 คำอธิบายยูสเคสไดอะแกรมของการ Record file property to file .....	39
4.12 คำอธิบายยูสเคสไดอะแกรมของการ Get Access Permission .....	40
4.13 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ urllist.xml.....	50
4.14 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ filter.xml.....	51
4.15 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ preference.xml.....	52
4.16 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ alertMessage.xml.....	52

# สารบัญรูป

รูปที่	หน้า
2.1	ขั้นตอนการทำงานของ Contract Net Protocol ..... 7
2.2	FIPA-Compliant Agent Platform ..... 9
2.3	สถาปัตยกรรมหลักของเจดแพลตฟอร์ม..... 10
2.4	สถาปัตยกรรมระบบมัลติเอเจนต์สำหรับการจัดการทรัพยากรในระบบแบบกระจาย..... 11
2.5	การเปรียบเทียบประสิทธิภาพการทำงานระหว่างระบบแบบศูนย์กลางและแบบกระจาย.....13
2.6	GHDO, SHDO Template และ SHDO ..... 14
2.7	ซีเควนซ์ไคอะแกรมของระบบ GHDO-based MAS ..... 16
3.1	หน้าลงทะเบียนของเว็บไซต์ Watch That Page ..... 19
3.2	หน้าจอการจัดการรายชื่อเว็บไซต์ ของ Watch That Page ..... 19
3.3	หน้าลงทะเบียนของเว็บไซต์ The Web Watcher ..... 20
3.4	หน้าจอการจัดการรายชื่อเว็บไซต์ของ The Web Watcher ..... 20
3.5	หน้าจอการตั้งค่าคัดกรองของ The Web Watcher ..... 21
3.6	หน้าจอการตั้งช่องทางการแจ้งเตือนของ The Web Watcher ..... 22
3.7	รายงานระบบเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์ของ The Web Watcher ..... 22
3.8	หน้าจอการเพิ่มรายชื่อเว็บไซต์ของโปรแกรม WebMon ..... 23
3.9	หน้าจอการตั้งค่าความถี่การเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลของโปรแกรม WebMon ..... 23
3.10	หน้าจอการตั้งค่าคำคัดกรองของโปรแกรม WebMon ..... 24
3.11	หน้าจอการแจ้งเตือนของโปรแกรม WebMon ..... 24
3.12	หน้าจอการแสดงคุณสมบัติต่างๆของโปรแกรม WebMon ..... 25
4.1	สถาปัตยกรรมระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บ..... 28
4.2	ยูสเคสไคอะแกรมของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์..... 30
4.3	หน้าบันทึกรายการเว็บไซต์..... 31
4.4	หน้าบันทึกคำหลักและข้อมูลที่ใช้ในการคัดกรองข้อมูล..... 33
4.5	แจ้งเตือนผ่านแอปพลิเคชัน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง ..... 34
4.6	แจ้งเตือนผ่านอีเมล เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง..... 34
4.7	หน้าบันทึกการตั้งค่าช่องทางในการส่งข้อความเตือน เมื่อข้อมูลมีการเปลี่ยนแปลง..... 35
4.8	หน้าบันทึกการตั้งค่าข้อความเตือน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง ..... 36
4.9	หน้าแสดงรายงานการเปลี่ยนแปลงข้อมูล..... 37

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.10 ยูสเคสไดอะแกรม Web Fetcher Agent .....	37
4.11 ยูสเคสไดอะแกรม Server Agent .....	39
4.12 คลาสไดอะแกรม .....	41
4.13 สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์จัดลำดับงาน.....	43
4.14 สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เฝ้าสังเกต.....	44
4.15 สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์.....	46
4.16 สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์แจ้งเตือน.....	47
4.17 ซีควเอนซ์ไดอะแกรมของการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล.....	48
4.18 ซีควเอนซ์ไดอะแกรมของการเรียกดูรายงาน.....	49
5.1 ส่วนต่อประสานของเจดแพลตฟอร์ม.....	55
5.2 เอเจนต์ของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์บนเจดแพลตฟอร์ม.....	58

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันข้อมูลข่าวสารในอินเทอร์เน็ตมีปริมาณมาก และมีแนวโน้มว่าจะเพิ่มขึ้นและเปลี่ยนแปลงอย่างรวดเร็ว ผู้ใช้สามารถเข้าถึงข้อมูลเหล่านี้ทำได้หลายวิธี ไม่ว่าจะเป็นการเข้าถึงข้อมูลเหล่านั้น โดยตรงผ่านเว็บไซต์ หรือการเข้าถึงโดยผ่านเครื่องมือในการค้นหาข้อมูลต่างๆ ซึ่งข้อมูลข่าวสารในปริมาณมากนี้ หากมีการเปลี่ยนแปลงอยู่ตลอดเวลา อาจทำให้เกิดปัญหา กับผู้ใช้ในเรื่องของการติดตามข่าวสารได้ล่าช้า ในปัจจุบันมีการพัฒนาระบบที่ใช้ในการเฝ้าสังเกต การเปลี่ยนแปลงข้อมูลบนเว็บไซต์ในหลายรูปแบบ เช่น ระบบคอร์ปัสของบริษัทปิซิเนส ออนไลน์เป็นเว็บไซต์ที่มีบริการในการติดตามข่าวสารและสามารถตั้งค่าการแจ้งเตือนข่าวอัปเดต ของเว็บไซต์ที่ผู้ใช้สนใจผ่านทางอีเมล ระบบ WatchThatPage ของบริษัท ATS Consulting AS เป็นต้น ซึ่งลักษณะการใช้งานของระบบประเภทนี้จะอยู่ในรูปแบบของเว็บแอปพลิเคชันที่ผู้ใช้ จะต้องเข้าไปทำการสมัครสมาชิก และทำการบันทึกรายชื่อเว็บไซต์ที่ต้องการให้ระบบเฝ้าสังเกต จากนั้นเมื่อเว็บไซต์มีการเปลี่ยนแปลงระบบจะทำการแจ้งเตือนผู้ใช้ผ่านอีเมล ซึ่งการแจ้งเตือนใน ลักษณะนี้หากผู้ใช้ไม่ได้ทำการตรวจสอบอีเมลจะทำให้ผู้ใช้ไม่ได้รับทราบข่าวสารได้ในทันที

โครงการนี้เกิดจากแนวคิดที่ต้องการสนับสนุนการใช้งานด้านการติดตามข้อมูลข่าวสาร บนเว็บไซต์ที่ผู้ใช้สนใจได้รวดเร็วและทันต่อเหตุการณ์ ซึ่งช่วยให้ผู้ใช้งานมีความสะดวกและ ประหยัดเวลาในการเข้าถึงข้อมูลเหล่านั้น โดยจะพัฒนาระบบเป็นลักษณะของเอเจนต์แอปพลิเคชันด้วยภาษาจาวาและติดตั้งไว้ภายในเครื่องของผู้ใช้ มีเอเจนต์เคลื่อนที่ทำหน้าที่เป็นตัวแทนใน การเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลในแต่ละเว็บไซต์ที่ผู้ใช้กำหนด และทำการแจ้งเตือนให้ ผู้ใช้ทราบในทันทีผ่านช่องทางที่ผู้ใช้เลือก ไม่ว่าจะเป็นการแจ้งเตือนผ่านเอเจนต์แอปพลิเคชัน หรือการแจ้งเตือนผ่านอีเมล

โครงการศึกษาและพัฒนาระบบเฝ้าสังเกตข้อมูลบนเว็บไซต์ด้วยมัลติเอเจนต์ มี วัตถุประสงค์เพื่อใช้เป็นเครื่องมือในการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลข่าวสารต่างๆที่ผู้ใช้ สนใจ หากข้อมูลมีการเปลี่ยนแปลง ระบบจะทำการแจ้งให้ผู้ใช้ทราบ พร้อมทั้งมีรายงานของ ข้อมูลที่มีการเปลี่ยนแปลงในหน้าเว็บที่ผู้ใช้สามารถเรียกดูผ่านแอปพลิเคชันได้ โดยระบบจะ พัฒนาด้วยหลักการทำงานของมัลติเอเจนต์ในรูปแบบของเอเจนต์ที่ทำงานร่วมกัน นอกจากนี้ ระบบจะออกแบบเพื่อช่วยลดปริมาณการสื่อสารในเครือข่าย โดยในสถานการณ์เฝ้าสังเกตการ

เปลี่ยนแปลงของข้อมูล ระบบจะใช้รูปแบบการทำงานแบบเอเจนต์เคลื่อนที่มาช่วยในการทำงาน ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มัลติเอเจนต์ในระบบประกอบไปด้วยเอเจนต์มากกว่าหนึ่งตัวติดต่อสื่อสารกัน โดยเอเจนต์แต่ละตัวจะมีหน้าที่แตกต่างกัน โดยจะมีการแบ่งงานของระบบออกเป็นส่วนย่อยๆ ให้เอเจนต์แต่ละตัวมีหน้าที่รับผิดชอบและทำงานร่วมกัน การทำงานลักษณะนี้มีข้อดีคือ เอเจนต์แต่ละตัวมีขนาดเล็กและการทำงานแต่ละงานไม่ขึ้นต่อกัน ช่วยทำให้การทำงานทำได้อย่างรวดเร็ว เอเจนต์ทุกตัวจะร่วมกันทำงานเพื่อให้ได้ตามเป้าหมายที่กำหนดไว้

เอเจนต์เคลื่อนที่ จะสามารถเคลื่อนที่จากเครื่องหนึ่ง ไปยังอีกเครื่องหนึ่งบนเครือข่ายตามที่ระบบกำหนด เพื่อไปประมวลผลงานตามที่ได้รับมอบหมายจนสำเร็จ จากนั้นเอเจนต์จะนำเฉพาะผลลัพธ์ที่ได้กลับมารายงานผลยังต้นทาง ในส่วนนี้จะช่วยลดปริมาณการสื่อสารในเครือข่ายได้เป็นอย่างมาก

## 1.2 ขอบเขตของระบบ

ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์เป็นระบบที่สร้างขึ้นเพื่อตอบสนองความต้องการของผู้ใช้ที่ต้องการเข้าถึงข้อมูลที่เปลี่ยนแปลงในอินเทอร์เน็ตได้อย่างรวดเร็ว และทันต่อเหตุการณ์ โดยขอบเขตของงานมีดังนี้

1. การบันทึก แก้ไข รายชื่อเว็บไซต์
2. ตั้งค่าคำหลัก กลุ่มคำหลัก หรือวลี ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลบนเว็บไซต์
3. ตั้งค่าการติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์
4. ตั้งค่าช่องทางการส่งข้อความเตือน เพื่อแจ้งให้ผู้ใช้ทราบเมื่อระบบตรวจพบข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง
5. การบันทึก แก้ไข ข้อความเตือนที่ผู้ใช้ต้องการตั้งค่า เพื่อให้ระบบส่งมาแจ้งเตือนเมื่อระบบตรวจพบข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง
6. รายงานผลการเปลี่ยนแปลงของข้อมูลในหน้าเว็บ
7. ออกแบบการทำงานของแอปพลิเคชันให้สามารถฝังตัวใน System Tray ของวินโดวส์ได้

## 1.3 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะผู้ใช้งานระบบจะได้รับ จากระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เพิ่มประสิทธิภาพ และช่วยประหยัดเวลาในการเข้าถึงเว็บเพจที่มีการเปลี่ยนแปลงในอินเทอร์เน็ต โดยการนำเทคโนโลยีแบบมัลติเอเจนต์และเอเจนต์เคลื่อนที่มาใช้ในการพัฒนาระบบในส่วนต่างๆ
2. ช่วยลดปริมาณการสื่อสารในเครือข่าย โดยนำเทคโนโลยีแบบเอเจนต์เคลื่อนที่มาใช้ในการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูล
3. ทำให้สามารถติดตามข้อมูลข่าวสารที่สนใจได้อย่างรวดเร็ว และทันต่อเหตุการณ์
4. เพิ่มช่องทางเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์

#### 1.4 ขั้นตอนการพัฒนาาระบบ

ขั้นตอนในการดำเนินการพัฒนาโครงการประกอบด้วยขั้นตอนการทำงานต่างๆ ดังนี้

1. ศึกษางานวิจัยที่เกี่ยวข้อง และทำความเข้าใจเกี่ยวกับเทคโนโลยีการทำงานแบบมัลติเอเจนต์และเทคโนโลยีของเอเจนต์แบบเคลื่อนที่
2. ศึกษาเทคโนโลยีและเครื่องมือที่สนับสนุนในการสร้างพัฒนาระบบ
3. ศึกษาการทำงานของระบบแบบเดิม เช่น ระบบการเฝ้าสังเกตข้อมูลที่เป็นแบบเว็บแอปพลิเคชัน พร้อมทั้งทำการวิเคราะห์ปัญหาและข้อจำกัดของระบบเดิม จากนั้นนำเอาแนวคิดที่ศึกษามาช่วยในการออกแบบระบบใหม่
4. กำหนดขอบเขต วิเคราะห์ และออกแบบระบบงานใหม่ด้วยเทคโนโลยีมัลติเอเจนต์ โดยแสดงด้วยแผนภาพดังต่อไปนี้
  - ยูสเคสไดอะแกรม แสดงภาพรวม และหน้าที่ของระบบ พร้อมทั้งแสดงปฏิสัมพันธ์ระหว่างผู้ใช้งานกับระบบ และระหว่างเอเจนต์หลักกับระบบ
  - แอกทิวิตีไดอะแกรม แสดงลำดับของกิจกรรมและสถานะของขั้นตอนการทำงานที่เกิดขึ้นภายในระบบ
  - คลาสไดอะแกรม แสดงความสัมพันธ์ระหว่างคลาสต่างๆภายในระบบ รวมถึงคุณสมบัติ และการดำเนินการที่สามารถกระทำกับคลาสนั้นๆได้
  - ซีควเอนซ์ไดอะแกรม จำลองกระบวนการทำงานที่เป็นกิจกรรมของระบบ ซึ่งกิจกรรมหนึ่งๆ นั้นเกิดจากการที่อ็อบเจกต์หนึ่ง ได้ตอบกับอีกอ็อบเจกต์หนึ่ง ด้วยการรับส่งข้อความกัน
  - สเตทชาร์ทไดอะแกรมแสดงถึงการเปลี่ยนสถานะการทำงานของเอเจนต์แต่ละตัวภายในระบบ

5. พัฒนาเอเจนต์ที่เป็นส่วนประกอบต่างๆของระบบ และประกอบรวมกันให้สามารถ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการดำเนินงานแบบมัลติเอเจนต์เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ทำการจำลองการทำงานของเซิร์ฟเวอร์ และทดสอบการทำงานของระบบ
7. จัดทำเอกสารสรุปผล และเอกสารคู่มือ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# เทคโนโลยีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎี เทคโนโลยีที่เกี่ยวข้องในการพัฒนาระบบ และงานวิจัยที่เกี่ยวข้อง ซึ่งเนื้อหาในบทนี้จะกล่าวถึงเทคโนโลยีเอเจนต์ เอเจนต์เคลื่อนที่ และกระบวนการทำงานแบบมัลติเอเจนต์ รวมถึงคุณลักษณะและสถาปัตยกรรมของ Java Agent Development Framework ดังต่อไปนี้

### 2.1 เอเจนต์

เอเจนต์เป็น โปรแกรมที่มีความสมบูรณ์ในตัวเอง มีคุณสมบัติที่สำคัญคือสามารถทำงานได้เองโดยอัตโนมัติ สามารถกระทำการต่างๆ ได้โดยไม่ต้องถูกสั่งงานจากผู้ใช้โดยตรง ควบคุมการกระทำของตนเองได้ สามารถแก้ปัญหาบางอย่างที่เกิดขึ้นได้เองและทำงานได้อย่างรวดเร็ว มีพฤติกรรมการเรียนรู้ การทำงานร่วมกันสามารถที่จะติดต่อสื่อสารระหว่างเอเจนต์เพื่อให้บริการ และร้องขอรับบริการจากเอเจนต์อื่นได้ สามารถโต้ตอบกับผู้ใช้ได้ โดยเอเจนต์แบ่งออกเป็น 7 ประเภทคือ (เทอดพร สุโพธิ์ และคณะ. 2552)

1. เอเจนต์ที่ทำงานร่วมกัน (Collaborative Agent) เป็นเอเจนต์ที่สามารถควบคุมการทำงานของตนเองและสามารถทำงานร่วมกับเอเจนต์ตัวอื่นๆ เพื่อให้เป็นไปตามเป้าหมายที่กำหนด การทำงานของเอเจนต์ชนิดนี้จะอาศัยเทคนิคการเจรจาต่อรองกันเพื่อให้เกิดเป็นการทำงานร่วมกัน ซึ่งเราเรียกกลุ่มของเอเจนต์ที่ทำงานร่วมกันลักษณะนี้ว่ามัลติเอเจนต์

2. เอเจนต์ที่ติดต่อกับผู้ใช้ (Interface Agent) ทำหน้าที่ติดต่อและอำนวยความสะดวกให้กับผู้ใช้ นอกจากนี้ เอเจนต์จะทำการเรียนรู้พฤติกรรมของผู้ใช้ เพื่อที่จะนำมาช่วยเหลือและตอบสนองการทำงานกับผู้ใช้ต่อไป

3. เอเจนต์แบบเคลื่อนที่ (Mobile Agent) เอเจนต์ชนิดนี้จะสามารถเคลื่อนที่จากเครื่องหนึ่งไปยังอีกเครื่องหนึ่งตามเครือข่าย และสามารถทำงานได้ในระบบเครือข่ายหรือภายในสภาพแวดล้อมที่มีความหลากหลายทางด้านฮาร์ดแวร์ และซอฟต์แวร์ โดยภายในตัวเอเจนต์นั้นจะประกอบไปด้วยหมายเลขประจำตัว ชุดคำสั่งในการทำงาน สถานะการทำงานที่เกี่ยวข้องกับการทำงานของเอเจนต์ และข้อมูลที่เป็น และส่วนต่อประสานที่ไว้สำหรับติดต่อกับเอเจนต์ตัวอื่นๆหรือติดต่อกับสภาพแวดล้อมที่เอเจนต์เคลื่อนที่ไปทำงาน เพื่อประมวลผลงานตามที่ได้รับมอบหมายจนสำเร็จ จากนั้นเอเจนต์จะนำผลลัพธ์กลับมารายงานผลยังต้นทาง เอเจนต์เคลื่อนที่จะมีขนาดเล็ก และการทำงานในลักษณะนี้จะช่วยลดปริมาณการสื่อสารของข้อมูลในเครือข่ายได้

4. เอเจนต์ข้อมูล (Information Agent) ทำหน้าที่จัดการข้อมูลข่าวสารที่กระจายอยู่ตามแหล่งต่างๆบนเครือข่ายแบบกระจาย โดยส่วนใหญ่จะใช้ในการค้นหา และสำรวจข้อมูลบนอินเทอร์เน็ตให้ตรงกับความต้องการของผู้ใช้มากที่สุด

5. เอเจนต์ที่โต้ตอบ (Reactive Agent) ทำหน้าที่ตอบสนองต่อสิ่งกระตุ้นภายใต้สภาพแวดล้อมที่เอเจนต์นั้นอยู่

6. เอเจนต์แบบผสม (Hybrid Agent) คือเอเจนต์ที่พัฒนาขึ้นมาโดยรวบรวมเฉพาะข้อดีของเอเจนต์แต่ละแบบมารวมกันได้

7. เอเจนต์แบบประกอบกัน (Heterogeneous Agent) คือเอเจนต์ที่มีความหลากหลายทั้งในด้านหน้าที่และลักษณะในการทำงานตั้งแต่ 2 ชนิดขึ้นไป เอเจนต์เหล่านี้จะทำงานร่วมกัน มีการติดต่อสื่อสารเพื่อแลกเปลี่ยนข้อมูลกัน ซึ่งจะอาศัยการติดต่อสื่อสารด้วยภาษากลางของเอเจนต์ที่เรียกว่า ACL (Agent Communication Language)

## 2.2 มัลติเอเจนต์

การทำงานแบบมัลติเอเจนต์จะประกอบไปด้วยเอเจนต์มากกว่าหนึ่งตัวทำการติดต่อสื่อสารกันและทำงานร่วมกัน โดยเอเจนต์แต่ละตัวในระบบจะมีคุณสมบัติหรือหน้าที่เหมือนหรือแตกต่างกันก็ได้ แต่เอเจนต์ทุกตัวจะร่วมกันทำงานเพื่อให้ได้ตามเป้าหมายที่กำหนดไว้ โดยมีกลไกการทำงานที่สำคัญ 2 ประการคือ (Bellifemine, et al. 2007)

1. การประสานงาน (Coordination) เป็นการประสานงานกันระหว่างเอเจนต์เพื่อทำให้เกิดการทำงานที่เชื่อมโยงกัน สอดคล้องกัน เนื่องจากเอเจนต์แต่ละตัวที่ทำงานร่วมกันนั้นมีหน้าที่การทำงานที่เหมือนหรือแตกต่างกัน การประสานงานกันระหว่างเอเจนต์ช่วยจัดการในเรื่องโครงสร้างการทำงานภายในมัลติเอเจนต์ ทำการวางแผนการทำงานของมัลติเอเจนต์ เพื่อหลีกเลี่ยงการทำงานที่ไม่สอดคล้องหรือขัดแย้งกันระหว่างเอเจนต์ ช่วยจัดการเรื่องการสื่อสารและการเจรจาต่อรองกันระหว่างเอเจนต์

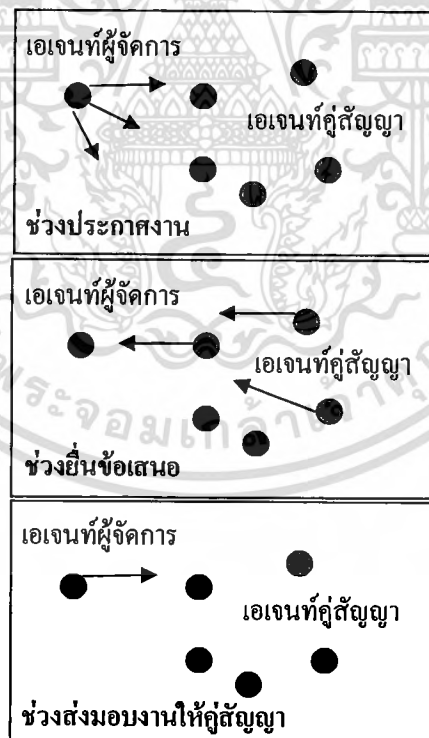
2. การสื่อสาร (Communication) เป็นกลไกการสื่อสารระหว่าง เอเจนต์ที่ทำงานร่วมกัน หรือติดต่อสื่อสารกับผู้ใช้และทรัพยากรต่างๆภายในระบบ เพื่อโต้ตอบและแลกเปลี่ยนข้อมูลข่าวสารซึ่งกันและกัน มีภาษากลางที่ใช้ในการสื่อสารคือ ACL โดยจะเน้นที่รูปแบบของข้อความที่ใช้สื่อสารกัน ทำให้อยู่ในรูปแบบที่เป็นมาตรฐานเดียวกัน และเอเจนต์ทุกตัวใช้รูปแบบข้อความเดียวกันตามที่ตกลง ซึ่ง ACL จะแบ่งออกเป็น 2 ประเภทคือ

- Knowledge Query and Manipulation Language (KQML) มีรูปแบบของภาษาสำหรับใช้ในการสื่อสารระหว่างเอเจนต์ที่มีลักษณะเป็นอ็อบเจกต์ คล้ายกับการเขียน โปรแกรมเชิงวัตถุ โดยรูปแบบของข้อความที่ใช้มีลักษณะเหมือนคลาสที่สามารถเพิ่มพารามิเตอร์ ในรูปของแอตทริ

บิวต์และค่าของแอตทริบิวต์ตามที่ต้องการได้ เอเจนต์แต่ละตัวจะมีชุดข้อความที่ใช้ในการโต้ตอบหรือสื่อสารกับเอเจนต์อื่นๆในระบบ สำหรับในชุดข้อความหนึ่งๆที่ประกอบไปด้วยแอตทริบิวต์ต่างๆนั้นเราเรียกว่า Performative

สำหรับรูปแบบของข้อความของ FIPA ACL จะคล้ายกับ KQML แต่ส่วนที่แตกต่างกันของภาษาทั้งสองแบบก็คือลักษณะของ Performative ที่กำหนดในแต่ละภาษา และความหมายของแต่ละ Performative ที่กำหนด

เทคนิคการสื่อสารของ FIPA ACL มีหลายวิธีการ และวิธีการที่ได้รับความนิยมคือ Contract Net Protocol หลักการทำงานจะเป็นระบบแบบกระจายการทำงาน โดยจะมีเอเจนต์ที่ทำงานร่วมกันอยู่ 2 ประเภท คือ เอเจนต์ที่เป็นผู้จัดการ (Manager) กับ เอเจนต์ที่เป็นคู่สัญญา (Contractor) โดยเอเจนต์ในระบบจะพยายามทำงานที่ได้รับมอบหมายมาด้วยทรัพยากรภายในระบบของตนเองก่อน ถ้าเอเจนต์ไม่สามารถทำงานหรือแก้ไขปัญหาหนึ่งๆได้ด้วยตนเอง เอเจนต์ในระบบจะทำการแตกงานหรือปัญหานั้นออกเป็นส่วนย่อยๆ และให้มีการจัดการแบ่งงานแต่ละส่วนให้กับกลไกการทำสัญญา และมีการแจกงานให้กับเอเจนต์ตัวอื่นเป็นผู้จัดการงานแทน โดยจะแสดงวิธีการทำงานดังรูปที่ 2.1



รูปที่ 2.1 ขั้นตอนการทำงานของ Contract Net Protocol

จากรูปที่ 2.1 ขั้นตอนการทำงานของ Contract Net Protocol มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่เอเจนต์ที่เป็นผู้จัดการจะทำหน้าที่ประกาศงานที่ต้องการให้เอเจนต์ตัวอื่นๆช่วยทำ ซึ่งด้านการคำนวณว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เอเจนต์คู่สัญญาจะทำการยื่นข้อเสนอการทำงานตามที่เอเจนต์ที่เป็นผู้จัดการประกาศไว้
- เอเจนต์ที่เป็นผู้จัดการจะประเมินข้อเสนอที่ได้รับมาจากเอเจนต์คู่สัญญา และทำการเลือกข้อเสนอที่ดีที่สุดและทำการส่งมอบงานให้กับเอเจนต์คู่สัญญานั้นทำงานต่อไป

รูปแบบการทำงานของมัลติเอเจนต์แบ่งเป็น 4 ประเภทหลักคือ (เทอดพร สุโพธิ์ และคณะ. 2552)

1. เอเจนต์ที่ทำงานร่วมกัน (Cooperating Agents) คือการแบ่งงานย่อยๆ ให้เอเจนต์แต่ละตัวรับผิดชอบและทำงานร่วมกัน การทำงานลักษณะนี้มีข้อดีคือ เอเจนต์แต่ละตัวมีขนาดเล็กและการทำงานแต่ละงานไม่ขึ้นต่อกัน แต่วิธีการนี้จะต้องมีการจัดการรูปแบบของการสื่อสารของเอเจนต์แต่ละตัวให้อยู่ในรูปแบบหรือมาตรฐานเดียวกัน เพื่อช่วยให้การสื่อสารทำได้อย่างรวดเร็วและมีความถูกต้องตรงกัน

2. การวางแผนการทำงานร่วมกัน (Multi-agent Planning) เป็นการให้เอเจนต์สามารถทำงานแบบมีแผนร่วมกัน คือให้เอเจนต์แต่ละตัวมีหน้าที่ต่างกัน แต่ทำงานให้บรรลุเป้าหมายอย่างเดียวกัน การทำงานลักษณะนี้จะมีเอเจนต์ตัวกลางที่ทำหน้าที่วางแผนให้เอเจนต์แต่ละตัวในทีมทำงานเข้าจังหวะกัน โดยให้เอเจนต์แต่ละตัวมีการสื่อสารแลกเปลี่ยนข้อมูลกันเพื่อทำให้เกิดการปรับปรุงข้อมูลและพฤติกรรมการทำงานที่เข้ากันได้ของเอเจนต์แต่ละตัวในทีม

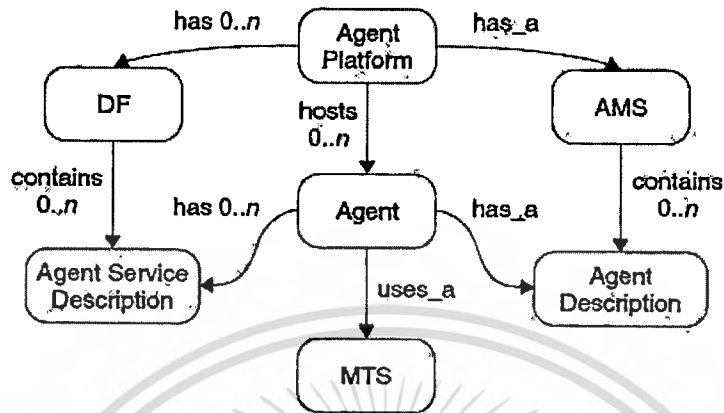
3. เอเจนต์ที่ทำงานแบบแข่งขันกัน (Competing Agents) คือการพัฒนาเอเจนต์หลายตัวที่ทำงานในลักษณะเดียวกันหรือทำหน้าที่เดียวกัน และมีการจัดการให้มีการแข่งขันกันทำงานอย่างใดอย่างหนึ่ง จากนั้นนำผลลัพธ์ที่ได้มาดูว่าผลลัพธ์ของเอเจนต์ตัวใดดีที่สุด มีประสิทธิภาพมากที่สุด และระบบจะนำผลลัพธ์นั้นมาใช้งานต่อไป

4. เอเจนต์ที่ทำงานแบบมีการเจรจาต่อรองกัน (Negotiation Agents) คือการพัฒนาเอเจนต์ให้มีลักษณะการประเมินการทำงานของเอเจนต์ในทีม และทำการเลือกเอเจนต์ตัวที่ดีที่สุดมาทำงาน โดยจะมี Contract Net Protocol เป็นพื้นฐานในการทำงาน ซึ่งการทำงานลักษณะนี้จะใช้ในระบบของการประมูลเป็นหลัก

### 2.3 เจด

เจด (JADE หรือ Java Agent Development Framework) คือซอฟต์แวร์แพลตฟอร์ม เป็นมิดเดิลแวร์ที่ช่วยในการพัฒนาระบบมัลติเอเจนต์ให้สามารถทำได้สะดวกและช่วยลดความยุ่งยากในการพัฒนา เนื่องจากเจดมีลักษณะการทำงานแบบระบบแบบกระจาย โดยเอเจนต์แต่ละตัวที่ทำงานบนแพลตฟอร์มนี้ จะทำงานในลักษณะของเธรด (Thread) ที่แต่ละตัวมีการทำงานที่เป็นอิสระจากกัน เจดเป็นโอเพนซอร์สที่พัฒนาด้วยภาษาจาวา ที่สนับสนุนการทำงานบนแพลตฟอร์มไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

J2ME นอกจากนี้ ในเจดแพลตฟอร์มมีการพัฒนาเอเจนต์ที่เป็นตามมาตรฐานของ FIPA ซึ่งมีองค์ประกอบต่างๆ ดังรูปที่ 2.2 (Bellifemine, et al. 2007)



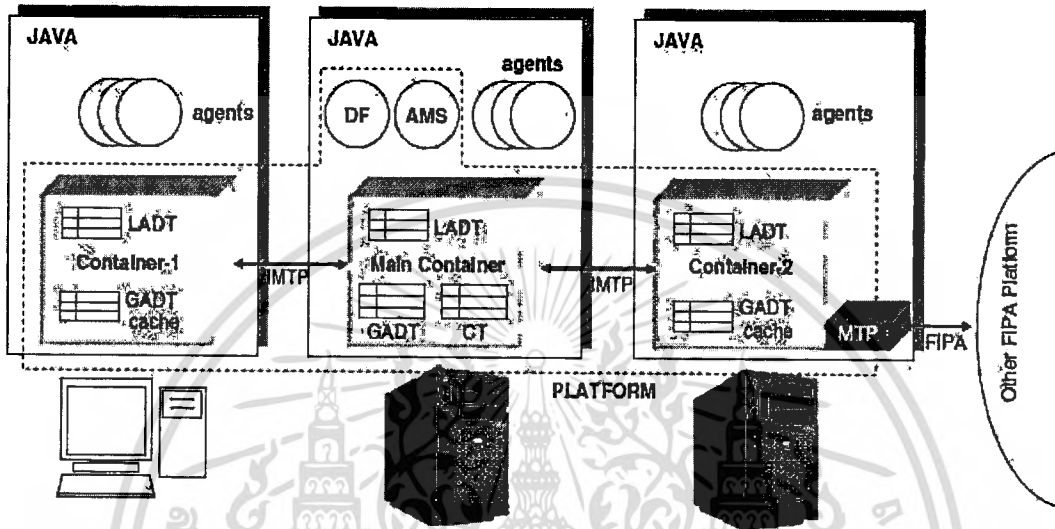
รูปที่ 2.2 FIPA-Compliant Agent Platform (Bellifemine, et al. 2007 :15)

องค์ประกอบของ FIPA-Compliant Agent Platform ในเจดเฟรมเวิร์คประกอบไปด้วย

- Agent Platform (AP) เป็นโครงสร้างหลักที่จะนำเอเจนต์มาติดตั้งและทำให้เอเจนต์ทำงานร่วมกัน
- Directory Facilitator (DF) เป็นเอเจนต์ที่เก็บรายชื่อเอเจนต์ที่อยู่ในแพลตฟอร์ม และให้บริการข้อมูลเกี่ยวกับเอเจนต์เหล่านั้น โดยเอเจนต์ที่เป็นผู้ให้บริการจะต้องมาทำการลงทะเบียนที่ Directory Facilitator เพื่อประกาศว่าตนเองให้บริการสิ่งใดบ้าง (Agent Service Description) สำหรับในแต่ละแพลตฟอร์มอาจจะมี Directory Facilitator หรือไม่ก็ได้
- Agent Management System (AMS) เป็นเอเจนต์ที่สามารถควบคุมการเข้าถึงและการใช้เอเจนต์ในแพลตฟอร์ม รวมทั้งเก็บสถานะและรายละเอียดของเอเจนต์ (Agent Description) โดยเอเจนต์แต่ละตัวจะต้องมาลงทะเบียนกับเอเจนต์ควบคุม เมื่อลงทะเบียนแล้วเอเจนต์แต่ละตัวจะได้รับหมายเลขประจำตัว (AID) ทั้งนี้ เพื่อเป็นการทำสัญญาและอนุญาตให้เอเจนต์ควบคุมสามารถจัดการควบคุมเอเจนต์เหล่านั้นได้ สำหรับในแต่ละแพลตฟอร์มจะมีเอเจนต์ควบคุมเพียงหนึ่งตัว
- Message Transport Service (MTS) ให้บริการในเรื่องของการควบคุมการสื่อสารและการรับส่งข้อความระหว่างเอเจนต์ที่อยู่ในแพลตฟอร์มเดียวกัน และสามารถใช้ในการสื่อสารข้ามแพลตฟอร์มได้ โดยข้อความที่ใช้ในการสื่อสารจะใช้มาตรฐานของ FIPA ACL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.3 แสดงสถาปัตยกรรมหลักของเจดแพลตฟอร์ม ซึ่งในแพลตฟอร์มจะไปประกอบด้วยเอเจนต์คอนเทนเนอร์ที่กระจายตัวอยู่บนเครือข่าย เอเจนต์ของระบบจะอยู่ในคอนเทนเนอร์ที่เป็นลักษณะของจาวาโพรเซส ภายในแพลตฟอร์มจะมีคอนเทนเนอร์หลัก (Main Container) ที่ทำหน้าที่จัดการและควบคุมคอนเทนเนอร์อื่นๆของระบบ



รูปที่ 2.3 สถาปัตยกรรมหลักของเจดแพลตฟอร์ม (Bellifemine, et al. 2007 :32)

โดยคอนเทนเนอร์ทุกตัวภายในระบบจะต้องมาทำการลงทะเบียนกับคอนเทนเนอร์หลัก และภายในคอนเทนเนอร์หลักจะมีส่วนประกอบดังนี้คือ

- Container Table (CT) ทำหน้าที่เก็บตำแหน่งอ้างอิงและตำแหน่งที่เอเจนต์ทำการเคลื่อนย้ายไปของคอนเทนเนอร์ทุกตัวในแพลตฟอร์ม
- Global Agent Descriptor Table (GADT) ทำหน้าที่เก็บสถานะและตำแหน่งปัจจุบันของเอเจนต์ที่อยู่ในแพลตฟอร์ม
- Local Agent Descriptor Table (LADT) ทำหน้าที่เก็บสถานะและตำแหน่งปัจจุบันของเอเจนต์ที่อยู่ในคอนเทนเนอร์

นอกจากนี้ ในคอนเทนเนอร์หลักยังมีเอเจนต์พิเศษ 2 ตัวหลักเป็นองค์ประกอบคือ Agent Management System และ Directory Facilitator ด้วย ซึ่งเป็นเอเจนต์ที่ใช้ควบคุมและจัดการเอเจนต์ทุกตัวในระบบ และภายในเจดแพลตฟอร์มยังมีเอเจนต์เฝ้าสังเกตระยะไกล (Remote Monitoring Agent หรือ RMA) ซึ่งเป็นเอเจนต์ที่มีการทำงานเป็นลักษณะของแอปพลิเคชัน ที่ให้ผู้ใช้ระบบสามารถใช้ในการตรวจสอบและติดตามการทำงานของเอเจนต์ต่างๆในระบบ

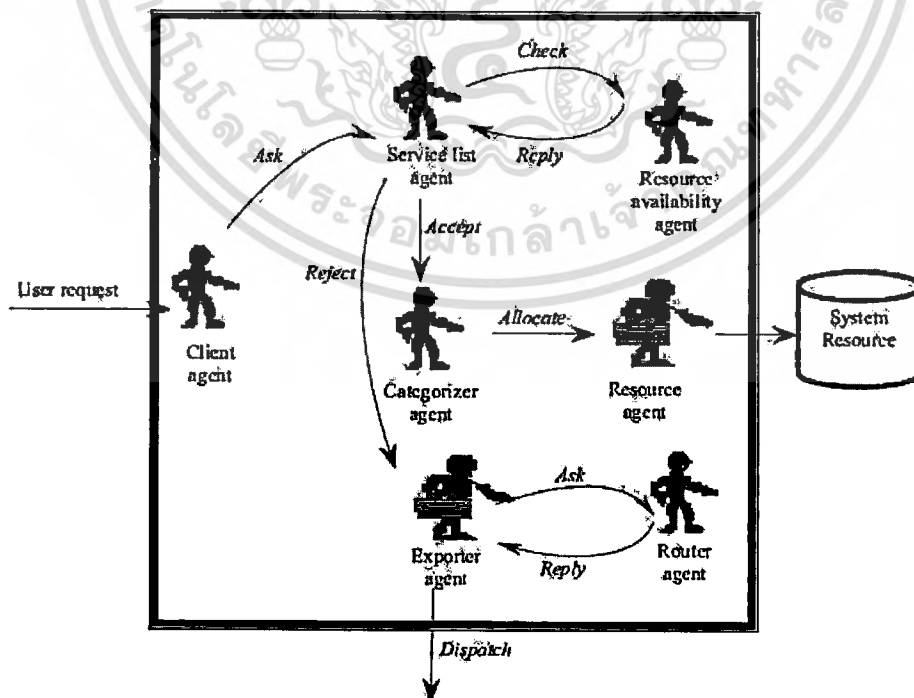
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 งานวิจัยที่เกี่ยวข้อง

### 2.4.1 การพัฒนาระบบมัลติเอเจนต์สำหรับการจัดการทรัพยากรในระบบแบบกระจาย

Kelash, et al. (2005) ได้พัฒนาระบบมัลติเอเจนต์แบบกระจาย เพื่อช่วยในการจัดการงานต่างๆ และเพื่อกระจายการจัดสรรทรัพยากรในระบบ เทคนิคในการบริหารจัดการระบบแบบกระจายส่วนใหญ่ จะเป็นลักษณะรวมศูนย์ ซึ่งเมื่อระบบแบบกระจายมีขนาดใหญ่ จะทำให้เกิดปัญหาตามมาคือ ทำให้เครือข่ายเกิดความคับคั่งขึ้นที่โหนดที่ทำหน้าที่เป็นตัวบริหารจัดการระบบ ส่งผลต่อประสิทธิภาพการทำงาน และเมื่อเกิดความผิดพลาดขึ้นที่โหนดดังกล่าว จะทำให้ระบบไม่สามารถทำงานได้ เป้าหมายหลักของระบบคือเพื่อลดการสื่อสารภายในเครือข่าย และช่วยให้การทำงานทำได้รวดเร็วขึ้น ดังนั้น ระบบแบบกระจายจึงต้องการแอปพลิเคชันที่นำมาใช้ในการบริหารจัดการทรัพยากรของระบบ เพื่อให้มีการใช้ทรัพยากรได้อย่างมีประสิทธิภาพ คุณสมบัติสำคัญของเอเจนต์ที่ช่วยในการจัดการทรัพยากรคือ สามารถทำงานได้เองโดยอัตโนมัติ เอเจนต์สามารถที่จะกระทำสิ่งต่างๆ ได้โดยไม่จำเป็นต้องถูกสั่งจากผู้ใช้โดยตรง สามารถที่จะควบคุมการกระทำของตนเองได้ สามารถแก้ปัญหาบางอย่างที่เกิดขึ้นได้เองและทำงานได้อย่างรวดเร็ว มีพฤติกรรมการเรียนรู้ การทำงานร่วมกัน และยังสามารถโต้ตอบกับผู้ใช้ได้

สถาปัตยกรรมของระบบมัลติเอเจนต์สำหรับการจัดการทรัพยากรในระบบแบบกระจาย ประกอบไปด้วยเอเจนต์ ทั้งหมด 7 ประเภท แสดงดังรูป 2.4



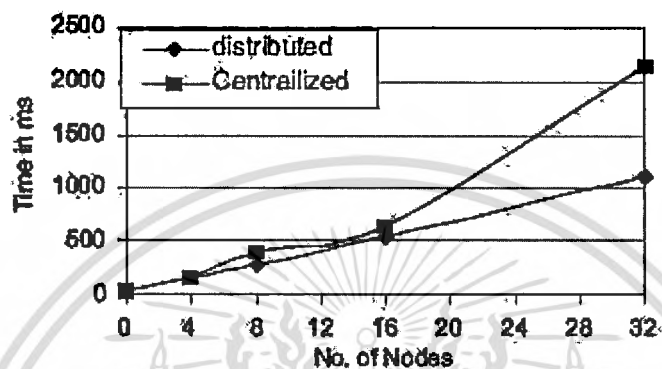
รูปที่ 2.4 สถาปัตยกรรมระบบมัลติเอเจนต์สำหรับการจัดการทรัพยากรในระบบแบบกระจาย

1. Client Agent ทำหน้าที่รับคำร้องขอจากผู้ใช้งาน สำหรับคำร้องขออาจจะเป็นการขอข้อมูลหรือสถานะข้อมูลของทรัพยากร หรืออาจจะเป็นการส่งข้อมูลหรือส่งคำสั่งเพื่อใช้จัดการบริหารทรัพยากร
2. Service List Agent ทำหน้าที่เก็บรายการของ Resource Agent โดยจะรับคำร้องขอมาจาก Client Agent และทำหน้าที่ส่งคำร้องขอนั้นให้กับ Resource Availability Agent ถ้ามีการตอบกลับว่าทรัพยากรภายในระบบว่าง ตัว Service List Agent จะส่งคำร้องขอนั้นให้กับ Categorizer Agent แต่ถ้าทรัพยากรภายในระบบไม่ว่าง Service List Agent จะทำการส่งคำร้องขอนั้นให้กับ Exporter Agent
3. Resource Availability Agent ทำหน้าที่แจ้งว่าทรัพยากรที่ต้องการว่างและพร้อมใช้งานหรือไม่ โดยจะรับคำร้องขอมาจาก Service List Agent และทำการตรวจสอบสถานะของทรัพยากร ตลอดจนเข้าถึงฐานข้อมูลที่เก็บสถานะของทรัพยากรในระบบ
4. Resource Agent ทำหน้าที่จัดการและควบคุมทรัพยากรของโหนด พร้อมทั้งทำการประมวลผลคำร้องขอ เอนเจนท์ตัวนี้จะเป็นเอนเจนท์เคลื่อนที่ เนื่องจากจะต้องเคลื่อนที่ไปมาระหว่างโหนดในระบบได้
5. Router Agent ทำหน้าที่จัดเตรียมเส้นทางในการเข้าถึงแหล่งทรัพยากรที่อยู่ภายนอกโหนด แต่อยู่ภายในเครือข่ายเดียวกันให้กับ Exporter Agent
6. Categorizer Agent ทำหน้าที่ของ Resource Agent ที่ว่างให้กับ Service List Agent
7. Exporter Agent เป็นเอนเจนท์เคลื่อนที่ที่ทำหน้าที่นำคำร้องขอของผู้ใช้งานผ่านทาง Router Agent ได้กำหนดไว้ไปยังโหนดที่ต้องการเข้าถึงทรัพยากร โดยจะส่ง Request Resource ID ไปให้ Router Agent และรอการตอบกลับจาก Router Agent ถ้า Router Agent ไม่มีข้อมูลเกี่ยวกับทรัพยากรที่ต้องการ Exporter Agent จะพยายามเข้าถึงทรัพยากรที่ต้องการภายในโหนดของตนเอง

ระบบถูกพัฒนาบน Java Virtual Machine (JVM) ด้วย Aglets Software Development Kit (ASDK) ของ IBM Aglets โดยใช้เครื่องคอมพิวเตอร์จำนวน 32 เครื่องเชื่อมต่อกันในระบบเครือข่าย เพื่อจำลองให้เป็นระบบแบบกระจาย และในแต่ละเครื่องจะมีมัลติเอนเจนท์รันอยู่ภายใต้ Aglets Platform

สำหรับกรณีศึกษาของระบบจะใช้ในการจัดการพอร์ตแบบขนาน (Parallel Port) ของคอมพิวเตอร์ โดยจะให้เอนเจนท์ในระบบทำการค้นหา รับ/ส่งข้อมูล และควบคุมข้อมูลที่ไปยังอุปกรณ์ต่อพ่วงที่ต่ออยู่กับพอร์ตแบบขนานของคอมพิวเตอร์ในเครือข่าย ซึ่งจะวัดประสิทธิภาพของระบบโดยดูจากเวลาที่ตอบสนองการทำงานต่างๆภายในระบบ

การวัดประสิทธิภาพของระบบดูจากระยะเวลาของการตอบสนองกันระหว่างการส่งคำร้องและการตอบกลับมาจาก Exporter Agent จากโหนดที่ต้องการ เป้าหมายหลักของระบบคือช่วยให้ระบบในเครือข่ายสามารถทำงานได้อย่างมีประสิทธิภาพและช่วยลดเวลาในการทำงานโดยรวมของระบบ ผลการวัดประสิทธิภาพเปรียบเทียบการทำงานระหว่างระบบแบบศูนย์กลางและแบบกระจายแสดงดังรูปที่ 2.5



รูปที่ 2.5 การเปรียบเทียบประสิทธิภาพการทำงานระหว่างระบบแบบศูนย์กลางและแบบกระจาย (Kelash, et al. 2005)

จากรูปที่ 2.5 เวลาที่ตอบสนองการทำงานของระบบแบบกระจายจะตอบสนองได้รวดเร็วเมื่อจำนวนโหนดในระบบเพิ่มมากขึ้น ซึ่งในกรณีนี้จะสามารถรองรับการเพิ่มขยายของระบบได้ดีกว่าระบบแบบศูนย์กลาง และข้อดีอีกข้อหนึ่งคือระบบสามารถทำงานภายใต้ระบบปฏิบัติการใดๆได้ เนื่องจากระบบพัฒนาด้วยภาษาจาวา

การจัดการทรัพยากรด้วยระบบแบบกระจายสามารถทำงานได้อย่างรวดเร็ว และมีประสิทธิภาพ ไม่ว่าจะเป็นการค้นหา หรือจัดการทรัพยากรในระบบ และเมื่อเปรียบเทียบกับการทำงานแบบศูนย์กลางแล้ว การทำงานแบบกระจายด้วยการทำงานแบบมัลติเอเจนต์ และเอเจนต์เคลื่อนที่จะช่วยให้ระบบสามารถเพิ่มขยายได้ในอนาคต และสามารถเคลื่อนที่ไปทำงานยังโหนดอื่นๆบนเครือข่ายได้อย่างรวดเร็ว

#### 2.4.2 การออกแบบระบบมัลติเอเจนต์สำหรับงานทางด้านการแพทย์

Hadzic, et al. (2009) ได้ออกแบบระบบมัลติเอเจนต์เพื่อใช้ในการเข้าถึงข้อมูลทางด้านการแพทย์ โดยมีการจำแนกโครงสร้างของโรคของมนุษย์ออกเป็น 4 มิติ คือ ประเภทของโรค อาการของโรค สาเหตุของโรค และการรักษาโรค การออกแบบระบบมัลติเอเจนต์นี้จะประกอบไปด้วยเอเจนต์ 4 ประเภทคือ Interface Agent, Manager Agent, Information Agent และ Smart Agent ซึ่งการที่เลือกเอเจนต์มาใช้ในการออกแบบระบบ เนื่องจากคุณสมบัติสำคัญของเอเจนต์ที่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

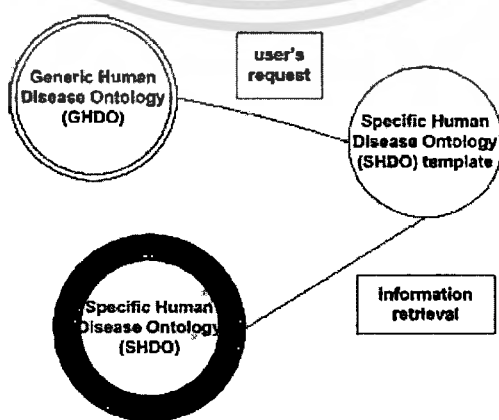
สามารถทำงานได้เองโดยอัตโนมัติ โดยไม่จำเป็นต้องถูกสั่งจากผู้ใช้หรือระบบอื่นๆ และเอเจนต์สามารถที่จะตัดสินใจและควบคุมการกระทำของเอเจนต์เองได้ เอเจนต์ยังสามารถทำงานร่วมกัน แบ่งปันข้อมูลกัน แก้ปัญหาที่ซับซ้อน และยังสามารถโต้ตอบกันเองได้

นอกจากนี้ ยังมีการนำแนวคิดเกี่ยวกับออนโทโลยีมาใช้ร่วมกับการออกแบบระบบมัลติเอเจนต์ เพื่อเข้าถึงข้อมูลทางการแพทย์ด้วย โดยออนโทโลยีจะเข้ามาช่วยสนับสนุนในเรื่องขององค์ความรู้ที่เกี่ยวข้องกับขอบเขตข้อมูลทางการแพทย์

Generic Human Disease Ontology (GHDO) คือออนโทโลยี เพื่อใช้ในการสนับสนุนการจัดการข้อมูล การใช้ข้อมูลร่วมกัน และการเข้าถึงข้อมูลทางการแพทย์ ภายในออนโทโลยีนี้จะประกอบไปด้วย ข้อมูลทั่วไปที่เกี่ยวข้องกับโรคของมนุษย์ โดยมองเป็นซูเปอร์เซตของโครงสร้างของโรคของมนุษย์ทั้ง 4 มิติ หรือ ซับออนโทโลยี (Subontology) ซึ่งประกอบไปด้วย ประเภทของโรค อาการของโรค สาเหตุของโรค และการรักษาโรค

Generic Human Disease Ontology (GHDO) – based MAS เป็นระบบมัลติเอเจนต์ ที่เอเจนต์แต่ละตัวทำงานในแต่ละโหนด เอเจนต์ทุกตัวในระบบจะสามารถทำงานร่วมกัน มีการเจรจาต่อรองกัน เพื่อให้เกิดการแลกเปลี่ยนข้อมูล องค์ความรู้ และบริการต่างๆ เพื่อให้บรรลุเป้าหมายการทำงานร่วมกัน

ออนโทโลยี GHDO ใช้สนับสนุนการเข้าถึงข้อมูลในโครงสร้างการทำงานแบบระบบมัลติเอเจนต์ ดังรูปที่ 2.6 โดยเมื่อนำมาใช้จะทำให้เกิดต้นแบบของ Specific Human Disease Ontology Template (SHDO) ซึ่งเป็นต้นแบบที่ผู้ใช้สนใจ เช่น ถ้าผู้ใช้สนใจเฉพาะสาเหตุของการเกิดโรค ตัวต้นแบบ SHDO จะมีเฉพาะซับออนโทโลยีที่เป็นสาเหตุของโรคเท่านั้น หลังจากการสืบค้นข้อมูล เอเจนต์จะทำการสร้างอินสแตนซ์เข้าไปที่ตัวต้นแบบ SHDO จากนั้น ตัวต้นแบบ SHDO จะทำการแปลงไป Specific Human Disease Ontology (SHDO) ซึ่งจะแสดงเป็นผลลัพธ์ให้กับผู้ใช้ต่อไป ภายใน SHDO จะประกอบไปด้วยข้อมูลที่เกี่ยวข้องกับโรคเฉพาะทาง เช่น Bipolar Disorder หรือ โรคที่มีความผิดปกติทางด้านอารมณ์ เป็นต้น



รูปที่ 2.6 GHDO, SHDO Template และ SHDO (Hadzic, et al. 2009 : 304)

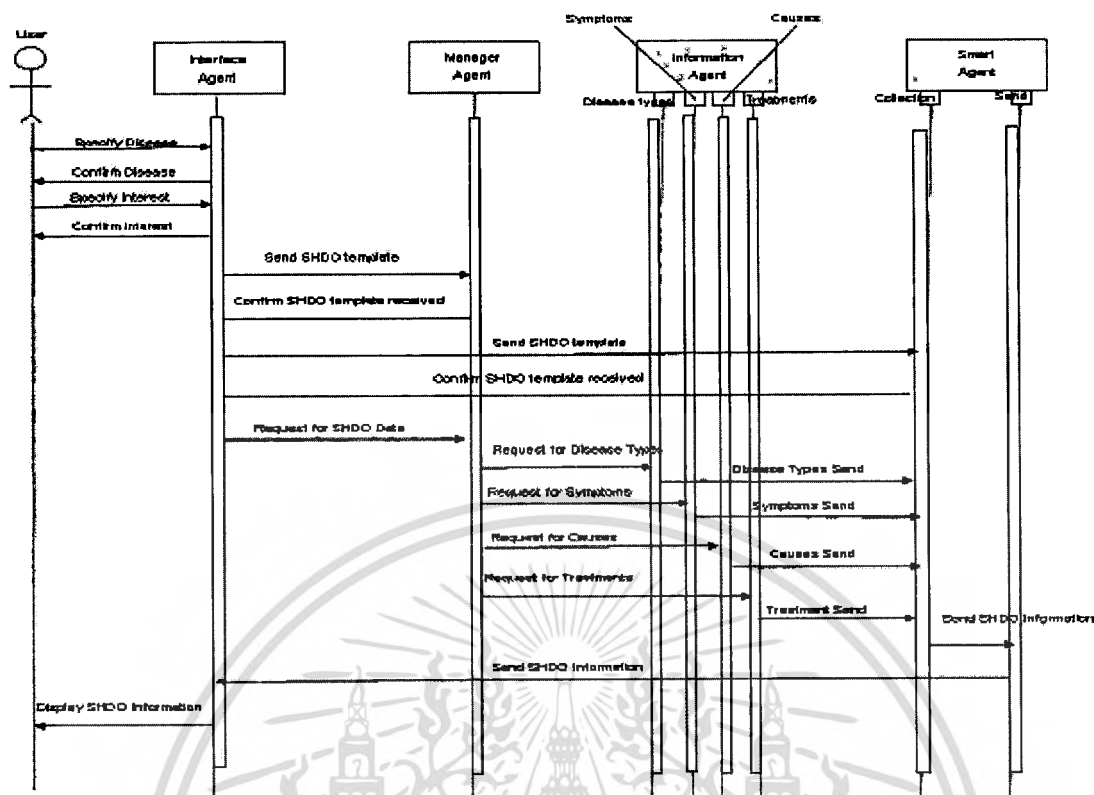
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในการทำงานของระบบนี้จะมอง Generic Human Disease Ontology เสมือนแหล่งข้อมูลหรือเส้นทางที่ช่วยในการค้นหาและเข้าถึงข้อมูลของเอเจนต์ที่อยู่ภายในระบบ ส่วนเอเจนต์จะทำหน้าที่ติดต่อกับผู้ใช้ สำหรับเอเจนต์ที่อยู่ภายในระบบมีลักษณะเอเจนต์นี้ แบ่งเป็น 4 ประเภทดังนี้

1. InterfaceAgent เป็นเอเจนต์ที่ทำหน้าที่สร้างตัวต้นแบบ SHDO จาก GHDO ซึ่งตัวต้นแบบนี้จะถูกส่งไปให้กับ ManagerAgent และ SmartAgent ตัวต้นแบบจะสอดคล้องกับการค้นหาของผู้ใช้ โดยจะมีข้อมูลที่ผู้ใช้สนใจประกอบอยู่ด้วย ในระหว่างกระบวนการเข้าถึงข้อมูล เอเจนต์ที่อยู่ในแต่ละโหนดจะช่วยกันนำข้อมูลที่เกี่ยวข้องมาใส่ไว้ในต้นแบบ
2. ManagerAgent ทำหน้าที่กำหนดภาระงานให้กับ InformationAgent โดย ManagerAgent จะต้องมียอดความรู้ที่เกี่ยวกับโครงสร้างของภาระงานต่างๆด้วย ซึ่งโครงสร้างของแต่ละภาระงานจะมาจากตัวต้นแบบ SHDO
3. InformationAgents ทำหน้าที่ดึงข้อมูลที่เอเจนต์ต้องการจากฐานข้อมูลทางการแพทย์จากหลายแหล่ง โดยจะดูจากความต้องการของเอเจนต์ว่าต้องการข้อมูลเกี่ยวกับ ประเภทของโรค อาการของโรค สาเหตุของโรค หรือการรักษาโรค จากนั้นจึงส่งข้อมูลที่ได้ออกไปให้กับ SmartAgent
4. SmartAgent จะทำการเก็บรวบรวมและวิเคราะห์ข้อมูลที่ได้รับมาจาก InformationAgent และทำการเลือกและรวบรวมเฉพาะข้อมูลที่เกี่ยวข้องกับโครงสร้างของตัวต้นแบบ SHDO จากนั้นจะทำการแปลงตัวต้นแบบ SHDO ไปเป็น Specific Human Disease Ontology และส่งกลับไปให้ InterfaceAgent เพื่อนำเสนอหรือตอบคำถามแก่ผู้ใช้

การนำเสนอโมเดล GHDO-Based MAS จะนำเสนอด้วยซีเควนซ์โคอะแกรม ดังรูปที่ 2.7 เพื่อแสดงการทำงาน บทบาทและหน้าที่ของเอเจนต์แต่ละตัวในระบบ หากเอเจนต์ตัวใดที่มีหน้าที่หรือบทบาทมากกว่า 1 อย่าง ในซีเควนซ์โคอะแกรมจะแสดงจำนวนเส้น lifeline ตามจำนวนของบทบาทและหน้าที่ของเอเจนต์นั้น ภายใต้ Composite คลาสเดียวกัน ดังรูปที่ 2.7 ประกอบไปด้วยเอเจนต์ 4 ตัว ตัวแรกคือ InterfaceAgent โดยครั้งแรกผู้ใช้ต้องการสอบถามข้อมูลเกี่ยวกับโรคที่ตนเองสนใจ โดยผู้ใช้จะทำการส่งข้อความ “Specify Disease” ไปให้ InterfaceAgent และ InterfaceAgent จะทำการยืนยันว่าได้รับข้อความ ขึ้นตอนต่อไป ผู้ใช้จะระบุเฉพาะส่วนหนึ่งของโรคที่ผู้ใช้สนใจเป็นพิเศษ ซึ่งจะเป็ข้อมูลที่เกี่ยวข้องแต่ละมิติของโรค คือ ประเภทของโรค อาการของโรค สาเหตุของโรค หรือการรักษาโรค ไปให้ InterfaceAgent และ InterfaceAgent จะทำการยืนยันว่าได้รับข้อความ จากนั้น InterfaceAgent จะทำการสร้างตัวต้นแบบ SHDO โดยใช้ Generic Human Disease Ontology ซึ่งลักษณะโครงสร้างของตัวต้นแบบ SHDO นี้จะสามารถให้เอเจนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ซีเควนซ์ไดอะแกรมของระบบ GHDO-based MAS (Hadzic, et al. 2009 : 306)

อื่นๆภายในระบบ สามารถนำข้อมูลที่เกี่ยวข้องมาใส่ในตัวต้นแบบได้ ตัวอย่างเช่น หากผู้ใช้ระบุว่าต้องการข้อมูลเกี่ยวกับโรคเบาหวาน และต้องการจะทราบเฉพาะข้อมูลเกี่ยวกับสาเหตุและอาการของโรคเท่านั้น โดยผู้ใช้ไม่ได้สนใจเกี่ยวกับการรักษาโรคนี้เลย

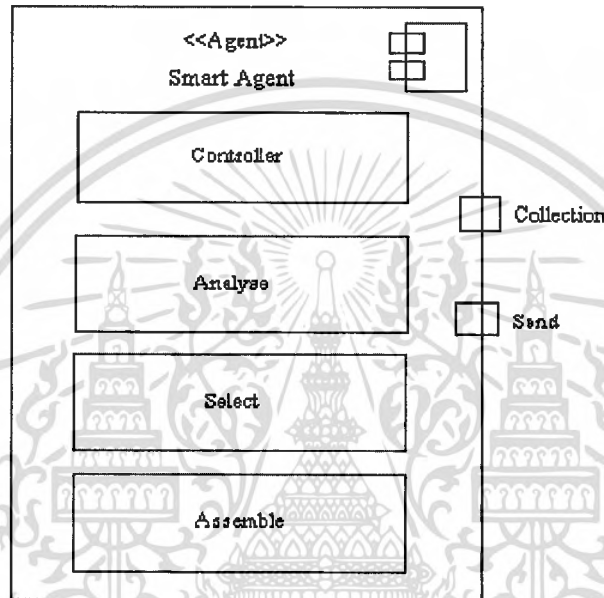
ตัวต้นแบบ SHDO จะถูกส่งไปให้ ManagerAgent ทันที เอเจนต์นี้จะทำหน้าที่เลือกว่าจะส่งคำร้องขอนี้ไปให้กับ InformationAgent ที่พอร์ตใดในระบบบ้าง ซึ่งที่ InterfaceAgent จะมีอยู่ด้วยกัน 4 พอร์ตที่ให้ข้อมูลเกี่ยวกับ ประเภทของโรค อาการของโรค สาเหตุของโรค หรือการรักษาโรค หลังจากนั้นตัวต้นแบบ SHDO จะถูกส่งต่อไปยัง SmartAgent พร้อมกับข้อมูลที่ได้มาจาก InformationAgent ในแต่ละส่วน ตัว SmartAgent จะทำหน้าที่วิเคราะห์ และรวบรวมข้อมูลในแต่ละส่งน เพื่อใส่กลับไปยังตัวต้นแบบ SHDO และส่ง SHDO นั้นกลับมายัง InterfaceAgent เพื่อให้ InterfaceAgent ส่งผลลัพธ์กลับมาให้ผู้ใช้

จากรูปซีเควนซ์ไดอะแกรม จะเห็นได้ว่าการออกแบบลักษณะของเอเจนต์โมเดลแบบ Composite Structure Diagram ที่ประกอบไปด้วยพาร์ท (Part) และ พอร์ต (Port) โดยแต่ละพาร์ทจะแสดงถึงขอบเขตการทำงานของเอเจนต์ และแต่ละพอร์ตแสดงถึงบทบาทแต่ละบทบาทของเอเจนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่น SmartAgent ดังรูปที่ 2.8 คือ Composite Structure Diagram ที่แสดงคุณสมบัติของ SmartAgent ที่ประกอบไปด้วยพอร์ต 2 พอร์ต แสดงบทบาทและหน้าที่ของ SmartAgent และพาร์ท 4 พาร์ท แสดงขอบเขตการทำงานที่อยู่ภายใน SmartAgent

โดยแต่ละพอร์ตจะใช้สำหรับติดต่อกับเอเจนต์ตัวอื่นๆ คือใช้ติดต่อกับ InterfaceAgent และ InformationAgent ในส่วนของการทำงานภายในตัว SmartAgent จะประกอบไปด้วย Controller, Analyze, Select และ Assemble



รูปที่ 2.8 Composite Structure Diagram แสดงคุณสมบัติของ SmartAgent (Hadzic, et al. 2009 : 307)

SmartAgent จะเก็บข้อมูลผ่านพอร์ต Collection ข้อมูลจะถูกนำมาวิเคราะห์ที่พาร์ท Analyze ข้อมูลจะถูกเลือกโดยพาร์ท Select เมื่อได้ข้อมูลที่ต้องการแล้ว จะมีการนำข้อมูลเหล่านั้นใส่กลับไปยังต้นแบบ SHDO ด้วยพาร์ท Assemble และ SHDO ที่ได้จะถูกส่งไปยัง InterfaceAgent ด้วยพอร์ต Send

## บทที่ 3

# การวิเคราะห์ระบบงานปัจจุบัน

การวิเคราะห์ระบบเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนหน้าเว็บไซต์ในปัจจุบัน ส่วนใหญ่ระบบที่พบจะอยู่ในลักษณะของเว็บแอปพลิเคชัน ดังนั้น กรณีศึกษาระบบงานปัจจุบันจึงทำการวิเคราะห์จากเว็บแอปพลิเคชันเหล่านั้น โดยเว็บไซต์ที่นำมาเป็นกรณีศึกษาจำนวน 2 เว็บไซต์ และ 1 แอปพลิเคชัน คือ เว็บไซต์ Watch That Page ([www.watchthatpage.com](http://www.watchthatpage.com)) และ เว็บไซต์ The Web Watcher ([www.thewebwatcher.com](http://www.thewebwatcher.com)) และ โปรแกรม WebMon ซึ่งมีรายละเอียดดังต่อไปนี้

### 3.1 เว็บไซต์ Watch That Page

Watch That Page เป็นเว็บไซต์ที่ให้บริการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์ และเก็บรวบรวมเนื้อหาของข้อมูลที่เปลี่ยนแปลงเหล่านั้นตามที่ผู้ใช้กำหนด โดยระบบแบ่งรูปแบบการบริการเป็น 2 รูปแบบคือ ไม่คิดค่าบริการ และแบบที่คิดค่าบริการ ซึ่งจะมีความแตกต่างกันที่จำนวนเว็บไซต์ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลง และรูปแบบในการแจ้งเตือนเมื่อข้อมูลมีการเปลี่ยนแปลง

ระบบจะนำเสนอข้อมูลที่มีการเปลี่ยนแปลงผ่านทางอีเมล หรือผ่านทางเว็บไซต์ส่วนบุคคลของผู้ใช้ระบบ โดยระบบมีบริการต่างๆดังนี้

1. ผู้ใช้ลงทะเบียนกับเว็บไซต์ ดังรูปที่ 3.1
  - ระบุข้อมูลส่วนตัวต่างๆ เช่น อีเมล ชื่อผู้ใช้ รหัสผ่าน เป็นต้น
  - ตั้งค่าการแจ้งเตือนหากข้อมูลมีการเปลี่ยนแปลง ผู้ใช้สามารถระบุได้ว่าต้องการให้ระบบตรวจสอบและแจ้งเตือนเป็นรายวัน หรือรายสัปดาห์
  - ตั้งเวลาเพื่อให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์
  - ตั้งค่าเนื้อหาที่ต้องการให้ระบบส่งอีเมลแจ้ง ผู้ใช้สามารถเลือกได้ว่าต้องการให้ระบบ ส่งข้อมูลที่มีการเปลี่ยนแปลงไปให้ทั้งหมด หรือส่งเฉพาะชื่อเว็บไซต์ที่มีการเปลี่ยนแปลง
  - การตั้งค่าคำหลักที่ต้องการให้ระบบเฝ้าสังเกตการเปลี่ยนแปลง โดยแบ่งเป็นคำหรือวลีที่ต้องพบในเนื้อหาของเว็บไซต์ หรือตั้งค่าเกี่ยวกับเว็บไซต์ที่เฝ้าสังเกตว่าต้องไม่มีคำที่ผู้ใช้กำหนด

2. ผู้ใช้สามารถเพิ่มรายชื่อเว็บไซต์ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล โดยผู้ใช้สามารถจัดกลุ่มหรือแยกประเภทให้กับเว็บไซต์เหล่านั้นได้ด้วย

เอกสารนี้เป็นเอกสารต้นฉบับที่กรายชื่อเว็บไซต์ตามกลุ่มที่กำหนดไว้ ดังรูปที่ 3.2 โดยขั้นตอนการดำเนินการดังกล่าวจะดำเนินการโดยอัตโนมัติ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Your monitor for changes on the web

# Watch That Page.com

Home

### Your profile

This information will be kept strictly CONFIDENTIAL and will NOT be shared with any other party or mailing list as confirmed in our terms and conditions

User name (email address) \*

User alias

Password \*

Repeat password \*

Last name \*

First name \*

Country \* Thailand

Time zone \* GMT +7 hours

Time zone description Thailand, Russian Fed zone 6

Account type Standard

[Advanced Settings...](#)

Email me changes  Daily

Weekly on

Halt email updates

hh:mm or hh:mm am/pm

Check for changes at 08:00  [Add update time](#)

Email content  Show me the changes

Report broken pages

Keyword match  Remove keyword matching

Match  the whole page

With all the words

With the exact phrase

At least one of the words

Without the words

Mondays  
 Tuesdays  
 Wednesdays  
 Thursdays  
 Fridays  
 Saturdays  
 Sundays

รูปที่ 3.1 หน้าลงทะเบียนของเว็บไซต์ Watch That Page

Your monitor for changes on the web

# Watch That Page.com

Home Profile Pages Settings Logout

### Your pages

[Add new](#) [What's changed?](#)

Here are the pages you are tracking via WatchThatPage. You can, at anytime, add or remove pages. When you are adding pages, there might be a small delay as we are downloading the initial page from the server.

Add page  to folder  Crane

Add folder  under  Top level

Your pages

---

Details  Collapse  Expand  View file  Move selected move to Crane

Crane

[www.lecmachinery.com](#)

Crane

Crane and Hoists service provider.

**Add pages while surfing!**  
 You can add the current page you are watching while you're surfing by setting up a special bookmark.  
[Read more about this >>](#)

click to select/unselect page or folder  
 click to open/close folder  
 page unreachable for a long time  
 page not downloaded last 24 hours

รูปที่ 3.2 หน้าการจัดการรายชื่อเว็บไซต์ ของ Watch That Page

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ที่ออกให้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 เว็บไซต์ The Web Watcher

The Web Watcher เป็นเว็บไซต์ที่ให้บริการฟรี เพื่อเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล หรือทรัพยากรอื่นๆ บนเว็บไซต์ตามที่ผู้ใช้กำหนด โครงสร้างการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลจะขึ้นอยู่กับสิ่งที่ผู้ใช้กำหนด ไม่ว่าจะเป็นคำหลักที่ผู้ใช้สนใจ หรือปริมาณการเปลี่ยนแปลงของข้อมูลเมื่อเทียบกับเนื้อหาของเว็บไซต์ ระบบจะมีการแจ้งเตือนตามช่องทางที่ผู้ใช้กำหนด โดยระบบมีบริการต่างๆ ดังนี้

1. ผู้ใช้ลงทะเบียนกับเว็บไซต์ ดังรูปที่ 3.3 โดยผู้ใช้ต้องระบุข้อมูลส่วนตัว เช่น ชื่อ อีเมล และรหัสผ่าน เป็นต้น

### TheWebWatcher

รูปที่ 3.3 หน้าลงทะเบียนของเว็บไซต์ The Web Watcher

2. ผู้ใช้สามารถเพิ่มรายชื่อเว็บไซต์ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล โดยผู้ใช้สามารถจัดกลุ่มหรือแยกประเภทให้กับเว็บไซต์เหล่านั้นได้ด้วย การตั้งชื่อกลุ่ม แล้วทำการบันทึกรายชื่อเว็บไซต์ตามกลุ่มที่กำหนดไว้ ดังรูปที่ 3.4

### TheWebWatcher

เอกสารนี้เป็นเอกสารรูปที่ 3.4 หน้าจอการจัดการรายชื่อเว็บไซต์ของ The Web Watcher ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ผู้ใช้สามารถตั้งค่าคัดกรอง เพื่อให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลตามสิ่งที่สนใจได้ ดังรูปที่ 3.5
- ความถี่ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์
  - คำ หรือวลี ที่ต้องการให้ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนท์
  - ตั้งค่าการติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์
  - จัดกลุ่มของ URL ตามความสนใจ เช่นกลุ่มเกี่ยวกับกีฬา การศึกษา เป็นต้น

## TheWebWatcher

The screenshot shows the 'The Web Watcher' interface with a 'FILTER MANAGEMENT' section. The interface is divided into several panels: 'Navigation', 'Shortcuts', 'Miscellaneous', and 'FILTER MANAGEMENT'. The 'FILTER MANAGEMENT' section includes fields for 'Filter name', 'Frequency' (set to 12 hours), and 'Triggers' (set to unlimited). It also has sections for 'Keyword Tracking' with options to track with all, exact phrase, or at least one of the words, and 'Watch for differences' with checkboxes for tracking differences and ignoring numeric changes. There are also fields for 'Resources to monitor' with a list of URLs and groups (Animals, Auctions, Breaking news, Computers, Literature).

รูปที่ 3.5 หน้าจอการตั้งค่าคัดกรองของ The Web Watcher

4. การตั้งค่าช่องทางการแจ้งเตือน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง ซึ่งระบบให้สามารถแจ้งเตือนได้เฉพาะอีเมลเท่านั้น แต่ไม่จำกัดจำนวนอีเมล โดยผู้ใช้สามารถ

กำหนดลักษณะข้อมูลที่ส่งไปได้ว่าเป็น Text/Plain หรือ HTML ดังรูปที่ 3.6 โฆษณาด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# TheWebWatcher

รูปที่ 3.6 หน้าจอการตั้งช่องทางการแจ้งเตือนของ The Web Watcher

5. ผู้ใช้สามารถเรียกดูรายงานที่ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์โดยระบุเงื่อนไขในการค้นหาตามเวลาที่กำหนดได้ ดังรูปที่ 3.7

# TheWebWatcher

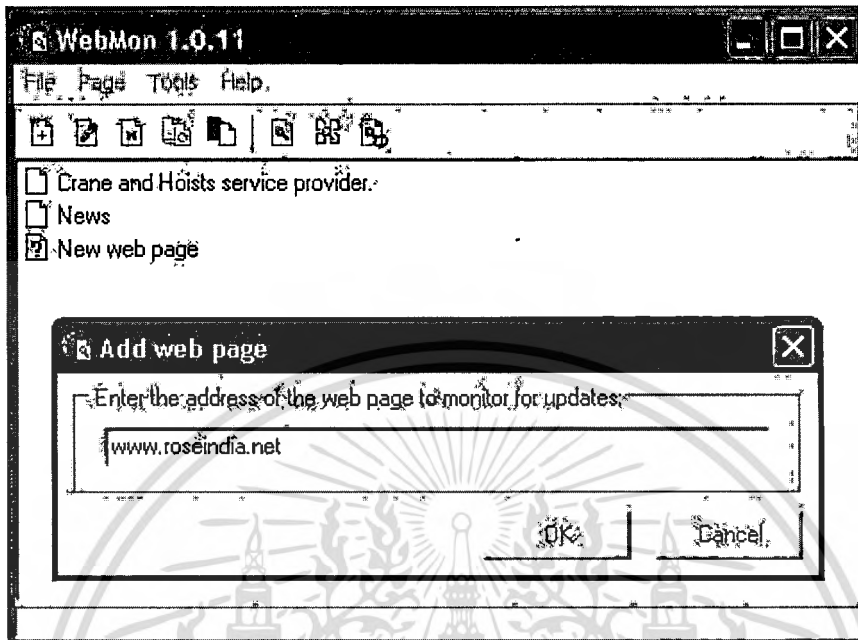
รูปที่ 3.7 รายงานระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ของ The Web Watcher

## 3.3 โปรแกรม WebMon

โปรแกรม WebMon เป็นแอปพลิเคชันที่รันบนวินโดวส์ สำหรับใช้ในการตรวจสอบการเปลี่ยนแปลงของข้อมูลบนเว็บไซต์แบบอัตโนมัติตามช่วงเวลาที่ผู้ใช้กำหนด เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง ระบบจะทำการแจ้งเตือนผ่านป๊อปอัพของระบบ โดยระบบจะมีบริการต่างๆ ดังนี้

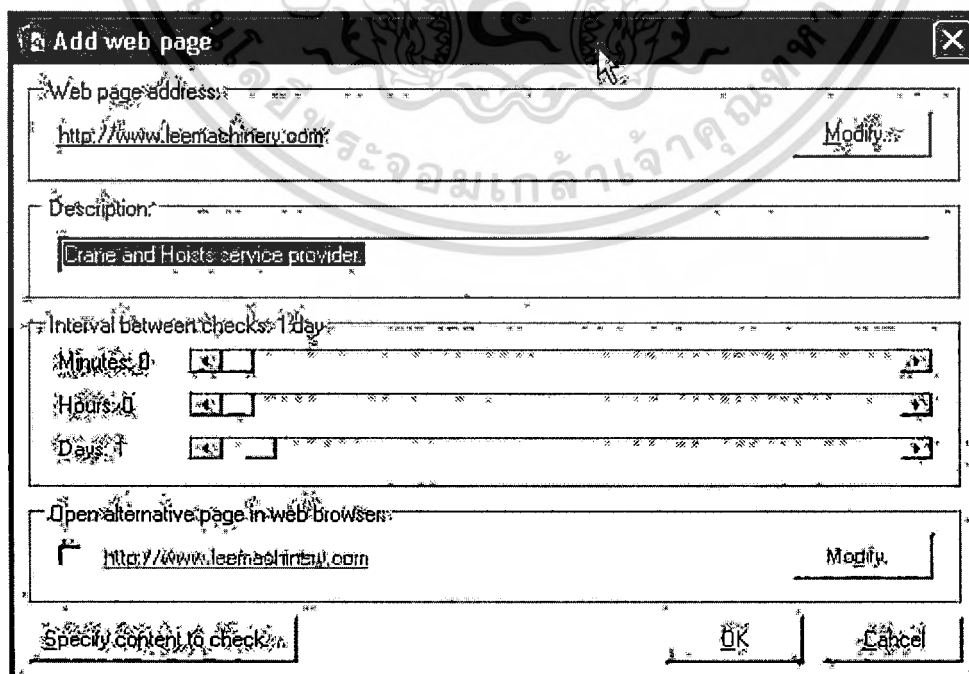
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ผู้ใช้สามารถเพิ่มรายชื่อเว็บไซต์ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล ดังรูปที่ 3.8



รูปที่ 3.8 หน้าจอการเพิ่มรายชื่อเว็บไซต์ของโปรแกรม WebMon

2. ระบบมีตัวเลือกเพื่อให้ผู้ใช้สามารถตั้งค่าเกี่ยวกับความถี่ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูล ดังรูปที่ 3.9



รูปที่ 3.9 หน้าจอการตั้งค่าความถี่การเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลของโปรแกรม WebMon

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาต

3. ผู้ใช้สามารถตั้งค่าช่วงของข้อมูลที่ต้องการจะเฝ้าสังเกตการเปลี่ยนแปลงของเว็บไซต์ตามช่วงที่สนใจได้ เช่น ผู้ใช้ตั้งค่าการเริ่มต้นให้ระบบเฝ้าสังเกตจะค้นไฟล์ ไปจนถึงข้อความบางส่วนที่อยู่ในไฟล์ได้ ดังรูปที่ 3.10



รูปที่ 3.10 หน้าจอการตั้งค่าคำคัดกรองของ โปรแกรม WebMon

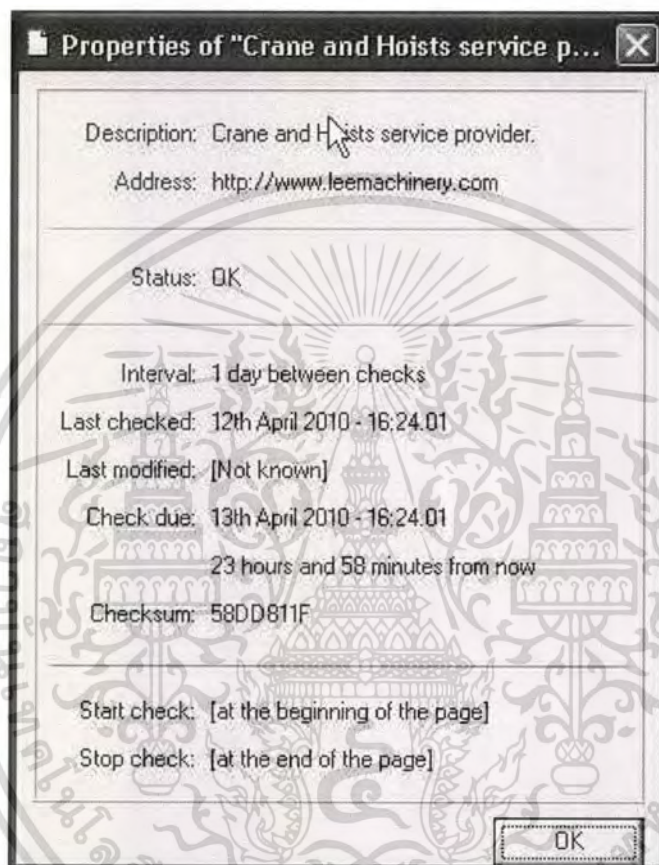
4. เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลงระบบจะแสดงการแจ้งเตือนผ่านหน้าต่างแบบผุดขึ้นของแอปพลิเคชัน ดังรูปที่ 3.11 เมื่อผู้ใช้คลิกที่หน้าต่างแบบผุดขึ้น ระบบจะเปิดการเชื่อมต่อไปยังเว็บไซต์นั้น โดยระบบจะเปิดโปรแกรมเว็บเบราว์เซอร์พร้อมทั้งแสดงหน้าเว็บไซต์นั้นขึ้นมา



รูปที่ 3.11 หน้าจอการแจ้งเตือนของ โปรแกรม WebMon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักผู้ดูแลเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ผู้ใช้สามารถเรียกดูรายละเอียดต่างๆ เกี่ยวกับการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์ที่กำหนดได้ ดังรูป 3.12 โดยระบบจะแสดงข้อมูลเกี่ยวกับวันเวลาที่ล่าสุดที่ระบบไปเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลของเว็บไซต์นั้น และบอกถึงวันเวลาที่การเฝ้าสังเกตครั้งต่อไป สถานะการตรวจสอบ และช่วงของข้อมูลที่ผู้ใช้ต้องการจะเฝ้าสังเกตการเปลี่ยนแปลงของเว็บไซต์ตามช่วงที่สนใจได้



รูปที่ 3.12 หน้าจอการแสดงผลคุณสมบัติต่างๆของโปรแกรม WebMon

### 3.4 ลักษณะการทำงานของระบบในปัจจุบัน

ระบบในปัจจุบันที่พบจะอยู่ในลักษณะของเว็บแอปพลิเคชัน ที่ให้ผู้ใช้ทำการลงทะเบียนเพื่อสมัครสมาชิกก่อนเข้าใช้งานระบบ และแบบแอปพลิเคชันบนวินโดวส์ที่มีการแจ้งเตือนเมื่อข้อมูลมีการเปลี่ยนแปลง แต่ไม่ได้แสดงการเปรียบเทียบว่ามีการเปลี่ยนแปลงในส่วนใดของหน้าเว็บเพจบ้าง สำหรับภายในระบบมีฟังก์ชันและขั้นตอนการทำงานดังต่อไปนี้

1. ผู้ใช้ทำการล็อกอินเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ผู้ใช้งานบันทึก แก้ไข URL ที่ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล
3. ตั้งค่าคำหลัก กลุ่มคำหลัก หรือวลี ที่ต้องการให้ระบบทำการเฝ้าสังเกต
4. ตั้งค่าการติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์
5. ผู้ใช้งานทำการจัดกลุ่มของ URL ตามความสนใจ เช่น กลุ่มเกี่ยวกับกีฬา การศึกษา เป็นต้น
6. ตั้งค่าอีเมล และพารามิเตอร์ต่างๆ เช่น ช่วงเวลาในการส่งข้อความเตือน เพื่อให้ระบบส่งข้อความเตือน

### 3.5 ปัญหาที่พบในระบบปัจจุบัน

1. ช่องทางการแจ้งข้อความเตือนผู้ใช้มีแค่ 2 ช่องทาง คือผู้ใช้ทำการล็อกอินเข้าสู่ระบบเพื่อตรวจสอบข้อมูลที่เปลี่ยนแปลง และการแจ้งเตือนผ่านอีเมล ปัญหาที่พบคือหากผู้ใช้ไม่ได้ตรวจสอบอีเมลตลอดเวลา หรือไม่ได้ล็อกอินเข้าสู่ระบบเพื่อตรวจสอบข้อมูล อาจทำให้เกิดการติดตามข่าวสารล่าช้า
2. เนื่องจากระบบเป็นเว็บแอปพลิเคชัน หากผู้ใช้งาน ณ ช่วงเวลานั้นมีผู้ใช้เป็นจำนวนมาก อาจทำให้ขั้นตอนการทำงานบางอย่างมีความล่าช้า เช่น ในขั้นตอนการบันทึก แก้ไข URL หรือการตั้งค่าพารามิเตอร์ต่างๆ

### 3.6 ความต้องการของระบบงานใหม่

#### 3.6.1 ความต้องการหลักของระบบ

ระบบสามารถทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลบนเว็บไซต์ โดยระบบมีฟังก์ชันการทำงานต่างๆดังนี้

1. ระบบสามารถบันทึก แก้ไข รายชื่อเว็บไซต์ตามที่ผู้ใช้งานต้องการได้
2. ระบบสามารถให้ผู้ใช้งานตั้งค่าคำหลัก กลุ่มคำหลัก หรือวลี ที่ต้องการให้ระบบทำการเฝ้าสังเกต และสามารถตั้งค่าการติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์
3. ตั้งค่าความถี่ หรือช่วงเวลาที่ต้องการให้ระบบทำการเฝ้าสังเกตข้อมูล
4. ระบบสามารถให้ผู้ใช้งานตั้งค่าช่องทางการส่งข้อความเตือน และทำการการบันทึก แก้ไข ข้อความเตือนตามที่ต้องการได้
5. ระบบมีส่วนของรายงานที่แสดงผลการเปลี่ยนแปลงของข้อมูลในหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูาตใ้หน้าไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.6.2 ความต้องการอื่นของระบบ

1. การทำงานของระบบเป็นลักษณะของแอปพลิเคชันที่สามารถฝังตัวใน System Tray ของวินโดวส์ได้
2. การทำงานของระบบมีความรวดเร็วและมีประสิทธิภาพ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

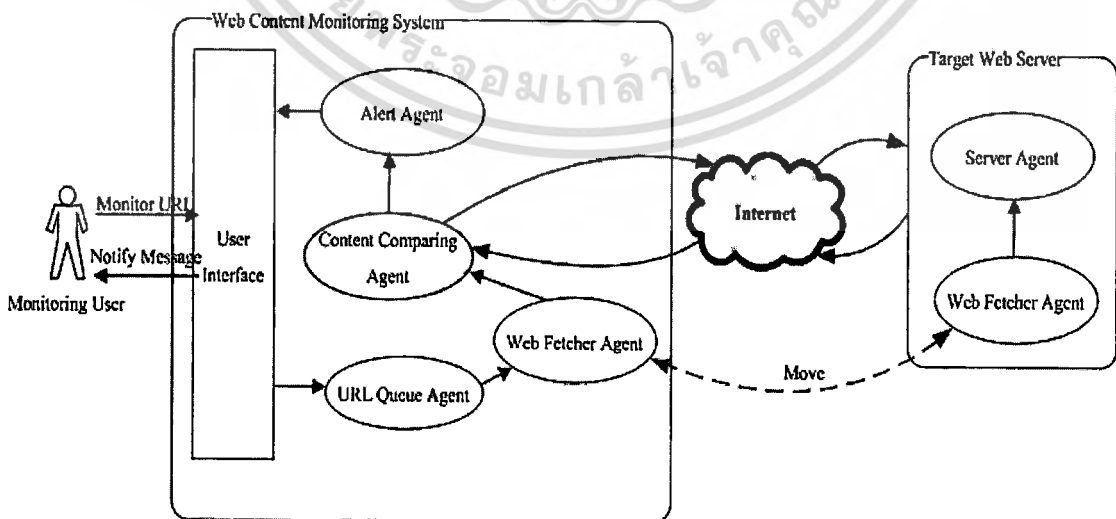
## บทที่ 4

# การวิเคราะห์และออกแบบระบบงานใหม่

หลังจากที่ได้ทำการวิเคราะห์ระบบแล้ว จึงทำการออกแบบระบบใหม่ขึ้นมา โดยนำเอาเทคโนโลยีการทำงานแบบมัลติเอเจนต์เข้ามาทำการพัฒนาระบบ และนำเอาเทคโนโลยีเอเจนต์เคลื่อนที่มาใช้ในส่วนของการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลเบื้องต้น เพื่อความรวดเร็วและลดปริมาณข้อมูลภายในเครือข่าย ซึ่งการออกแบบจะเน้นในเรื่องของความรวดเร็วในการรายงานผลการเปลี่ยนแปลงให้ผู้ใช้งาน โดยความต้องการและขอบเขตของระบบใหม่จะแสดงดังรายละเอียดต่อไปนี้

### 4.1 ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บ

การพัฒนาระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ เพื่อใช้เป็นเครื่องมือในการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลข่าวสารต่างๆที่ผู้ใช้สนใจ พร้อมทั้งทำการแจ้งให้ผู้ใช้งานและรายงานของข้อมูลที่มีการเปลี่ยนแปลงในหน้าเว็บที่ผู้ใช้สามารถเรียกดูผ่านแอปพลิเคชันได้ จะพัฒนาด้วยหลักการการทำงานของมัลติเอเจนต์ในรูปแบบของเอเจนต์ที่ทำงานร่วมกัน ซึ่งเอเจนต์แต่ละตัวจะทำหน้าที่แต่ละอย่างตามที่ระบบกำหนด และทำงานร่วมกันเพื่อให้บรรลุวัตถุประสงค์ของระบบ คือเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลในหน้าเว็บ รูปที่ 4.1 แสดงโครงสร้างการทำงานหลักของระบบ



รูปที่ 4.1 สถาปัตยกรรมระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับว่าผูกมัดให้ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเอเจนต์ในระบบจะประกอบไปด้วย

1. User Interface ทำหน้าที่ติดต่อกับผู้ใช้ และแสดงผลการรายงานการเปรียบเทียบผลการเปลี่ยนแปลงของข้อมูลที่ได้ทำการตรวจสอบมา
2. URL Queue Agent ทำหน้าที่จัดลำดับเว็บเพื่อให้ Web Fetcher Agent ไปทำการตรวจสอบหน้าเว็บ
3. Web Fetcher Agent เป็นเอเจนต์เคลื่อนที่ที่ทำหน้าที่ตรวจสอบข้อมูลหน้าเว็บเบื้องต้น เช่น วันที่แก้ไขล่าสุดของไฟล์ ขนาดไฟล์ข้อมูล โดยจะทำการตรวจสอบจากไฟล์ที่เอเจนต์เซิร์ฟเวอร์ได้ทำการประกาศไว้ที่เว็บเซิร์ฟเวอร์ ถ้าข้อมูลเหล่านี้มีการเปลี่ยนแปลง Web Fetcher Agent จะเคลื่อนที่กลับมาที่โคลเอนท์ เพื่อแจ้งให้ Content Comparing Agent ทราบ
4. Content Comparing Agent ทำหน้าที่ดึงข้อมูลมาจากเว็บเซิร์ฟเวอร์ และแยกส่วนที่เป็นเนื้อหาออกจากแท็กคำสั่งใน HTML พร้อมทั้งทำการเปรียบเทียบเนื้อหาของเว็บที่ได้มากับเนื้อหาที่บันทึกไว้ก่อนหน้า เมื่อข้อมูลเหล่านี้มีการเปลี่ยนแปลง Content Comparing Agent จะทำการแจ้งให้กับ Alert Agent ทราบ
5. Alert Agent ทำหน้าที่ส่งข้อความแจ้งเตือนผู้ใช้เมื่อเนื้อหาของเว็บมีการเปลี่ยนแปลงตามที่ผู้ใช้กำหนดเงื่อนไขต่างๆไว้ โดยช่องทางที่ระบบสามารถแจ้งเตือนได้มี 2 ช่องทาง คือ แจ้งเตือนผ่านแอปพลิเคชัน และแจ้งเตือนผ่านอีเมล
6. Server Agent จะอยู่บนเว็บเซิร์ฟเวอร์ ทำหน้าที่ให้บริการข้อมูลเกี่ยวกับคุณสมบัติของไฟล์ ได้แก่ วันที่แก้ไขล่าสุด ขนาดไฟล์

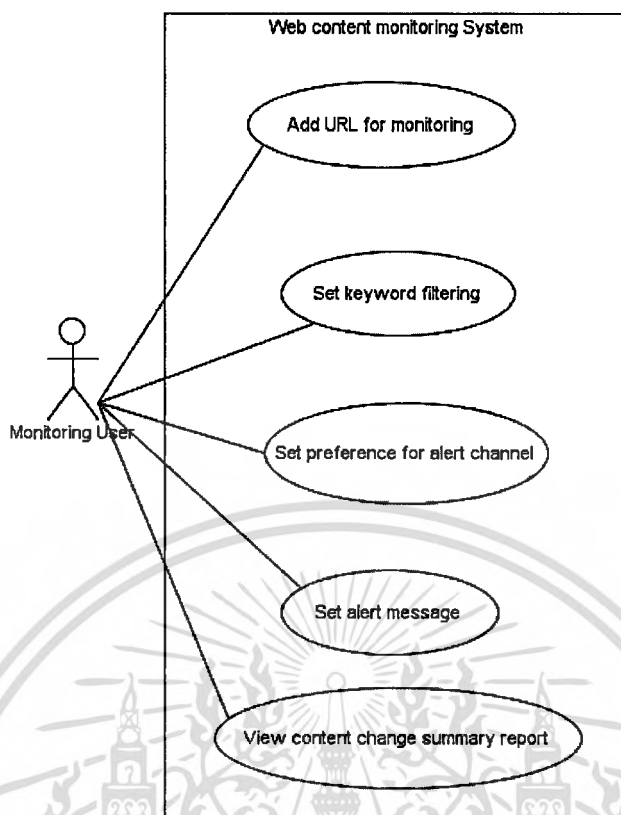
#### 4.2 ยูสเคสไดอะแกรม

ยูสเคสไดอะแกรมของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์ ทำให้ทราบถึงฟังก์ชันการทำงานของระบบ แสดงการปฏิสัมพันธ์กันระหว่างผู้ใช้งานกับระบบ ดังรูปที่ 4.2 โดยมีรายละเอียดคำอธิบายแอกเตอร์ ดังตารางที่ 4.1 และคำอธิบายยูสเคส ดังตารางที่ 4.2 ถึง 4.6 ตามลำดับ

ยูสเคสไดอะแกรมของ Web Fetcher Agent แสดงการปฏิสัมพันธ์กันระหว่าง Web Fetcher Agent และเอเจนต์อื่นๆในระบบ ดังรูปที่ 4.10 โดยมีรายละเอียดคำอธิบายแอกเตอร์ ดังตารางที่ 4.7 และ คำอธิบายยูสเคส ดังตารางที่ 4.8 ถึง 4.9 ตามลำดับ

ยูสเคสไดอะแกรมของ Server Agent แสดงการปฏิสัมพันธ์กันระหว่างเอเจนต์เคลื่อนที่กับเอเจนต์เซิร์ฟเวอร์ ดังรูปที่ 4.11 โดยมีรายละเอียดคำอธิบายแอกเตอร์ ดังตารางที่ 4.10 และ คำอธิบายยูสเคส ดังตารางที่ 4.11 ถึง 4.12 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ยูสเคสไดอะแกรมของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์

ตารางที่ 4.1 คำอธิบายแอกเตอร์ของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์

แอกเตอร์	รายละเอียด
Monitoring User	ผู้ใช้บริการที่ต้องการให้ระบบเฝ้าสังเกต ข้อมูลบนเว็บไซต์ที่ตนเองสนใจ

ตารางที่ 4.2 คำอธิบายยูสเคสไดอะแกรมของการ Add URL for monitoring

ยูสเคส	Add URL for monitoring
วัตถุประสงค์	เพื่อทำการบันทึก แก้ไขรายชื่อเว็บไซต์ที่ผู้ใช้ต้องการให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูล
แอกเตอร์ที่เกี่ยวข้อง	Monitoring User
สิ่งที่กระตุ้นการทำงาน	เลือกเมนู URL List
ขั้นตอนการทำงาน	1. Monitoring User ทำการกรอกชื่อเว็บไซต์ ที่อยู่เว็บไซต์ พร้อมทั้งเลือกตัวคัดกรองที่ต้องการ หากตัวคัดกรองไม่มีตามที่ต้องการ ผู้ใช้สามารถเพิ่มคำคัดกรองได้ที่เมนู Set keyword filtering

## ตารางที่ 4.2 (ต่อ)

	<p>2. ระบบทำการตรวจสอบการกรอกชื่อเว็บไซต์ว่าครบถ้วน ถูกต้องหรือไม่</p> <p>3. ทำการบันทึกหรือแก้ไขรายชื่อเว็บไซต์และรายละเอียดต่างๆ โดยสถานะการตรวจสอบเว็บไซต์เป็นเว็บไซต์ใหม่ (S) สถานะของเว็บไซต์มีดังนี้</p> <p>S = เว็บไซต์ใหม่</p> <p>C = เว็บไซต์ที่มีข้อมูลเปลี่ยนแปลง</p> <p>N = เว็บไซต์ไม่มีข้อมูลเปลี่ยนแปลง</p> <p>D = เว็บไซต์ที่เอเจนท์กำลังเคลื่อนที่ไปทำการเฝ้าสังเกต</p> <p>4. ระบบทำการบันทึกข้อมูลลงไฟล์ URL</p>
--	--

หน้าจอบันทึกรายการเว็บไซต์ (Add URL for monitoring) รูปที่ 4.3 แสดงหน้าจอที่กรอก รายชื่อเว็บไซต์ ที่ประกอบไปด้วยชื่อเว็บไซต์ ที่อยู่เว็บไซต์ วันที่ล่าสุดที่ระบบไปทำการเฝ้าสังเกต การเปลี่ยนแปลงบนเว็บไซต์นั้นๆ และสถานะของการตรวจสอบเว็บไซต์

Web Content Monitoring

File Preference Report About

URL List

Content Change		Site Name	URL	Last Monitor	Status
<input type="checkbox"/>		MThai	www.mthai.com	2553.03.23 ..	N
<input type="checkbox"/>		Posttoday	www.posttoday.com	2553.03.23 ..	N
<input type="checkbox"/>		IT	www.it.kmitl.ac.th	2553.03.23 ..	D

Delete

URL Detail

Site Name : Posttoday

URL : www.posttoday.com

Filtering : การเงิน

OK Clear

รูปที่ 4.3 หน้าจอที่กรอกเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 คำอธิบายยูสเคสโคอะแกรมของการ Set keyword filtering

ยูสเคส	Set keyword filtering
วัตถุประสงค์	เพื่อทำการตั้งกลุ่มตัวคัดกรอง ซึ่งประกอบไปด้วยการตั้งคำหลัก กลุ่มคำหลัก วลี และตั้งค่าการติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์ เพื่อให้ระบบทำการเฝ้าสังเกตการเปลี่ยนแปลงเฉพาะคำหลัก หรือปริมาณการเปลี่ยนแปลงของข้อมูลมาที่ผู้ใช้กำหนด
แอกเตอร์ที่เกี่ยวข้อง	Monitoring User
สิ่งที่กระตุ้นการทำงาน	เลือกเมนู Keyword Filtering
รายละเอียด	<p>1. Monitoring User ตั้งค่ารายละเอียดตัวคัดกรอง ดังนี้</p> <ul style="list-style-type: none"> <li>● ชื่อตัวคัดกรอง (Filter Name)</li> <li>● ระบุค่าความถี่ (Frequency) ที่ต้องการให้ระบบไปเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์</li> <li>● ระบุค่า (Keyword Filter) ที่ต้องการให้ระบบคัดกรองเมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง หากมีมากกว่า 1 คำ ให้เว้นวรรคแต่ละคำ เช่น หุ่น ดอกเบียร์ เป็นต้น ระบบจะนำคำว่าหุ่น และคำว่าดอกเบียร์ไปค้นหาในเว็บไซต์ ถ้าพบคำเหล่านี้ ระบบจะแจ้งให้ผู้ใช้ทราบ</li> <li>● ระบุวลี (Exact Phrase Filter) ต้องการให้ระบบคัดกรองเมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง เช่น “ดอกเบียร์เงินฝาก ธนาคารกรุงเทพ” ระบบจะนำวลีนี้ไปค้นหาในเว็บไซต์ที่มีการเปลี่ยนแปลง ถ้าพบวลีนี้ ระบบจะแจ้งให้ผู้ใช้ทราบ</li> <li>● ระบุเปอร์เซ็นต์การเปลี่ยนแปลงของข้อมูลเมื่อเทียบกับเนื้อหาทั้งหมดของเว็บไซต์ เช่น ระบุว่า 5% หมายถึง ถ้าข้อมูลมีการเปลี่ยนแปลงตั้งแต่ 5% ขึ้นไป ถึงจะแจ้งเตือนผู้ใช้</li> <li>● ระบุว่าถ้ามีการเปลี่ยนแปลงเฉพาะตัวเลข ผู้ใช้จะไม่สนใจในการเปลี่ยนแปลงนั้น ระบบไม่ต้องแจ้งเตือน (Ignore numeric change)</li> </ul> <p>2. ทำการบันทึกหรือแก้ไขรายละเอียดต่างๆ</p> <p>3. ระบบทำการบันทึกข้อมูลลง ไฟล์ filter.xml</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการใช้งาน ไม่ให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าจอบันทึกการตั้งค่าตัวคัดกรอง แสดงดัง รูปที่ 4.4 เพื่อนำไปใช้ในการเลือกกลุ่มตัวคัดกรองที่หน้าจอบันทึกการเว็บไซต์

	Filter Name	Frequency
<input type="checkbox"/>	อัตราดอกเบี้ย	5 Minutes
<input type="checkbox"/>	Java	5 Minutes
<input type="checkbox"/>	การเงิน	5 Minutes
<input type="checkbox"/>	ข่าวฉ้อ	5 Minutes

Filter Name :

Frequency :

Keyword Filter :

Exact Phrase Filter :

Alert if the new content at least  % of page content

Ignore numeric change

รูปที่ 4.4 หน้าบันทึกค่าหลักและข้อมูลที่ใช้ในการคัดกรองข้อมูล

ตารางที่ 4.4 คำอธิบายยูสเคสไคอะแกรมของการ Set alert channels

ยูสเคส	Set alert channels
วัตถุประสงค์	เพื่อทำการตั้งค่าช่องทางการส่งข้อความเตือนเมื่อเว็บไซต์มีการเปลี่ยนแปลงข้อมูลข่าวสาร เช่นตั้งค่าให้มีการเตือนผ่านแอปพลิเคชันของระบบ หรือตั้งค่าให้มีการเตือนผ่านอีเมล
แอกเตอร์ที่เกี่ยวข้อง	Monitoring User
สิ่งที่กระตุ้นการทำงาน	เลือกเมนูตั้งค่าช่องทางการส่งข้อความเตือน
รายละเอียด	<ol style="list-style-type: none"> <li>Monitoring User เลือกช่องทางการส่งข้อความเตือน</li> <li>ถ้าผู้ใช้เลือกช่องทางการส่งข้อความเตือนเป็นอีเมล ผู้ใช้ต้องทำการกรอกรายละเอียดของอีเมล</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### ตารางที่ 4.4 (ต่อ)

	<ol style="list-style-type: none"> <li>3. ระบบทำการตรวจสอบการกรอกรายละเอียดของอีเมลว่าครบถ้วน ถูกต้องหรือไม่</li> <li>4. เมื่อผู้ใช้คลิกปุ่ม Set ระบบทำการบันทึกข้อมูลลงไฟล์ preference.xml</li> <li>5. หากผู้ใช้คลิกปุ่ม Default ระบบจะตั้งค่าช่องทางการส่งข้อความเตือนผ่านแอปพลิเคชันเพียงอย่างเดียว</li> </ol>
--	---

หน้าจอบันทึกการตั้งค่าช่องทางการส่งข้อความเตือนแสดงดัง รูปที่ 4.7 ผู้ใช้สามารถบันทึกช่องทางการที่ต้องการให้ระบบส่งข้อความแจ้งเตือน โดยระบบจะสามารถให้ผู้ใช้ระบุได้ 2 ช่องทางคือ

- แจ้งเตือนผ่านแอปพลิเคชัน โดยการแสดงป๊อปอัพขึ้นมา ดังรูป 4.5



รูปที่ 4.5 แจ้งเตือนผ่านแอปพลิเคชัน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง

- แจ้งเตือนผ่านอีเมล โดยผู้ใช้ต้องระบุอีเมลที่ต้องการให้ระบบส่งข้อความไปแจ้งเตือน และรูปแบบอีเมลการแจ้งเตือนจะมีรูปแบบดังรูปที่ 4.6

#### URL Change alert

From: [webcontentmonitoring@gmail.com](mailto:webcontentmonitoring@gmail.com)

Sent: Monday, April 12, 2010 10:47:35 AM

To: maliwan\_lao@hotmail.com

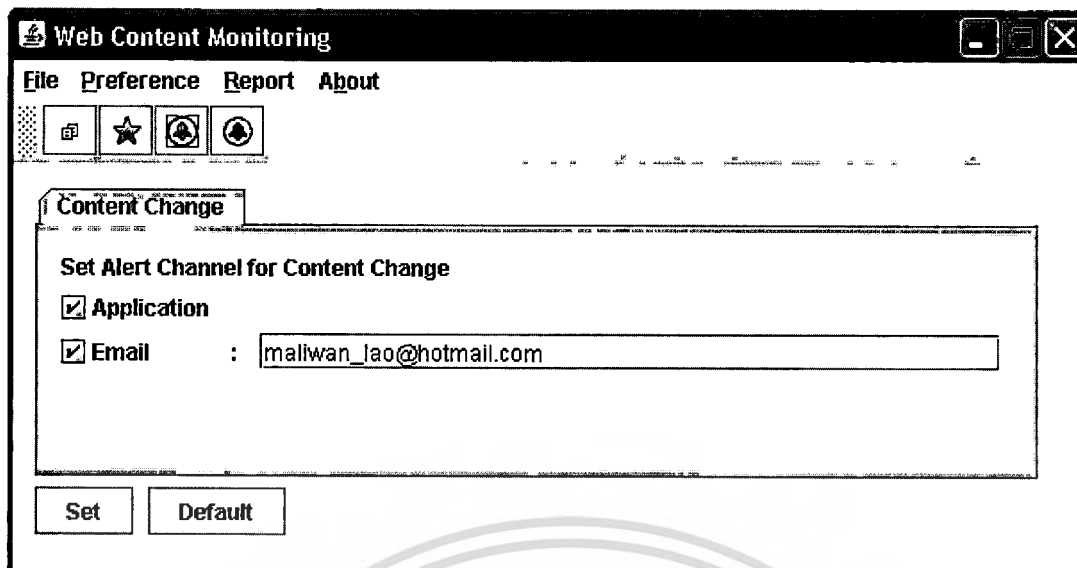
Content Change:Posttoday\_25530412\_105050.html

URL:www.posttoday.com

Monitor Date:2553.03.23 ??. at 10:58:40 GMT+07:00

รูปที่ 4.6 แจ้งเตือนผ่านอีเมล เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

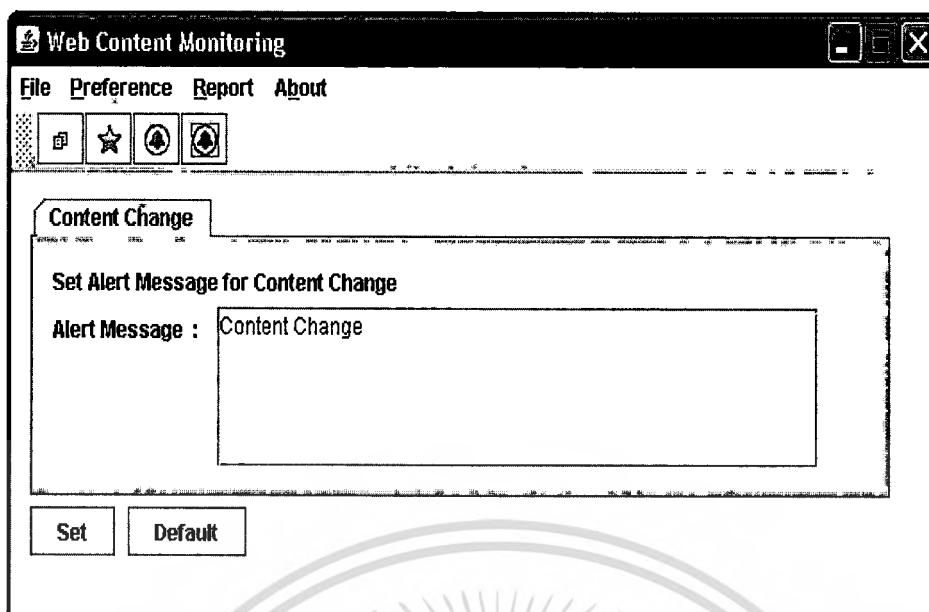


รูปที่ 4.7 หน้าจอบันทึกการตั้งค่าช่องทางในการส่งข้อความเตือน เมื่อข้อมูลมีการเปลี่ยนแปลง

ตารางที่ 4.5 คำอธิบายยูสเคสไดอะแกรมของการ Set alert message

ยูสเคส	Set alert message
วัตถุประสงค์	เพื่อทำการบันทึก หรือแก้ไขข้อความเตือนที่ระบบแจ้งผู้ใช้ เมื่อมีการเปลี่ยนแปลงเกิดขึ้นที่เว็บไซต์
แอกเตอร์ที่เกี่ยวข้อง	Monitoring User
สิ่งที่กระตุ้นการทำงาน	เลือกเมนูตั้งค่าข้อความเตือน
รายละเอียด	<ol style="list-style-type: none"> <li>1. Monitoring User กรอกข้อความเตือนสำหรับแจ้งเมื่อมีการเปลี่ยนแปลงเกิดขึ้นที่บนเว็บไซต์</li> <li>2. ระบบทำการตรวจสอบการกรอกข้อความเตือนว่าครบถ้วนหรือไม่</li> <li>3. ทำการบันทึกหรือแก้ไขข้อความเตือน</li> <li>4. ระบบทำการบันทึกข้อมูลลงไฟล์ alertMessage.xml</li> </ol>

รูปที่ 4.8 แสดงหน้าจอบันทึกการตั้งค่าข้อความเตือน ผู้ใช้สามารถตั้งค่าข้อความที่จะให้ระบบแจ้งเตือนเมื่อเว็บไซต์ที่ระบบเฝ้าสังเกตมีการเปลี่ยนแปลงข้อมูลเกิดขึ้น



รูปที่ 4.8 หน้าจอบันทึกการตั้งค่าข้อความเตือน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง

ตารางที่ 4.6 คำอธิบายยูสเคสไดอะแกรมของการ View content change summary report

ยูสเคส	Lookup content change summary report
วัตถุประสงค์	เพื่อเรียกดูรายงานข้อมูลที่เปลี่ยนแปลงบนหน้าเว็บไซต์
แอกเตอร์ที่เกี่ยวข้อง	Monitoring User
สิ่งที่กระตุ้นการทำงาน	เลือกเมนูรายงานข้อมูลที่เปลี่ยนแปลง
รายละเอียด	ระบบแสดงรายงานข้อมูลที่เปลี่ยนแปลงบนหน้าเว็บไซต์รายการปัจจุบัน

รูปที่ 4.9 แสดงหน้าจอรายงานการเปลี่ยนแปลงข้อมูลของเว็บไซต์ ผู้ใช้สามารถดูรายงานการเปลี่ยนแปลงได้ เมื่อเอเจนต์แจ้งเตือนได้ทำการแจ้งเตือนให้ผู้ใช้ทราบผ่านป๊อปอัพของแอปพลิเคชัน โดยในรายงานจะแสดงแถบสี เพื่อแสดงความแตกต่างของข้อมูลที่มีการเปลี่ยนแปลง โดยจะทำการเปรียบเทียบข้อมูลชุดเดิมก่อนหน้า กับข้อมูลล่าสุดที่เอเจนต์ได้ตรวจพบ ทางด้านซ้ายของรายงานจะเป็นข้อมูลเก่า ส่วนทางด้านขวาจะเป็นข้อมูลชุดใหม่ ระบบจะแสดงผลโดยการเปรียบเทียบบรรทัดต่อบรรทัดของข้อมูลแต่ละชุด

สำหรับข้อมูลที่ไม่มีการเปลี่ยนแปลงระบบจะไม่แสดงแถบสีใดๆ แต่ยังคงนำมาแสดงไว้ในหน้ารายงานเพื่อให้ผู้ใช้เห็น นอกจากนี้ ผู้ใช้สามารถกดปุ่ม Go to URL เพื่อทำการเชื่อมต่อไปยังเว็บไซต์ที่ระบบรายงานผลการเปลี่ยนแปลงของข้อมูล โดยระบบจะเปิดโปรแกรมเว็บเบราว์เซอร์พร้อมกับแสดงหน้าเว็บไซต์นั้นขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Web Content Monitoring**

File Preference Report About

**Content Change Summary Report**

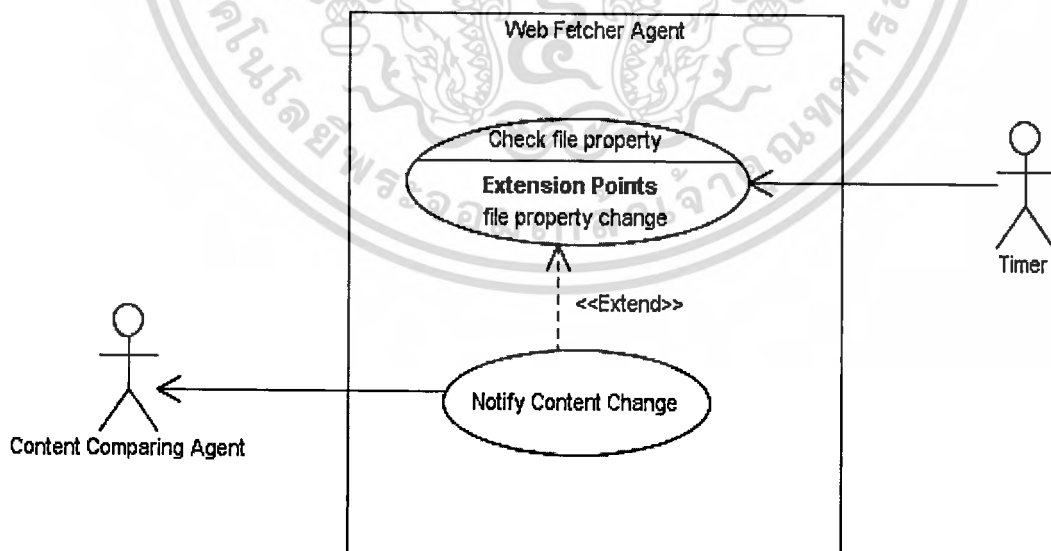
URL Address: www.posttoday.com

Monitor Date: 23/03/2553 พ.ศ. at 10:58:40 GMT+07:00

Old Content	New Content
latest breaking news headline	latest breaking news headline
ข่าวเด่น	ข่าวเด่น
เม้าท์กันให้ Z	เม้าท์กันให้ Z
ไทยคมยอมเชื่อมสัญญาณ ออกอากาศพีเพิลแซลแลแล้ว หลังส. ส.เพื่อไทยนำแดงกว่าพันคนบุกสถานี ลาซาลงแกวเจรจาขอคืนสัญญาณ	ไทยคมยอมเชื่อมสัญญาณออกอากาศ พีเพิลแซลแลแล้ว หลังส.ส.เพื่อไทยนำแดงกว่า พันคนบุกสถานีลาซาลงแกวเจรจาขอคืน สัญญาณ

Go To URL

รูปที่ 4.9 หน้าแสดงรายงานการเปลี่ยนแปลงข้อมูล



รูปที่ 4.10 ยูสเคสไดอะแกรม Web Fetcher Agent

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 แอจेंटที่เกี่ยวข้องกับ Web Fetcher Agent

แอจेंट	รายละเอียด
Content Comparing Agent	เอเจนต์ที่ทำหน้าที่ดึงข้อมูลจากเว็บไซต์มาเปรียบเทียบกับข้อมูลเดิม
Timer	กำหนดเวลาที่ทำให้ Web Fetcher Agent ไปทำการอ่านไฟล์ข้อมูลบนเว็บเซิร์ฟเวอร์

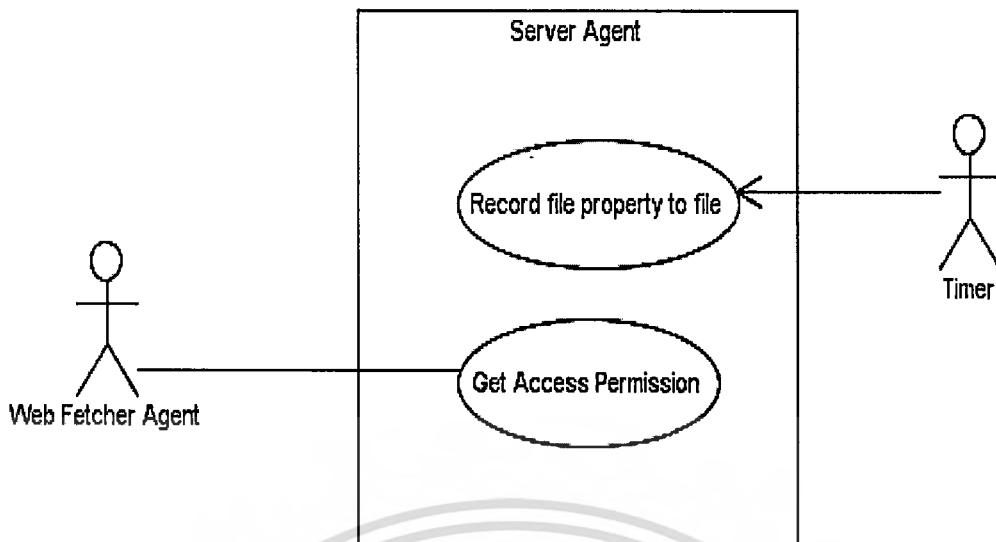
ตารางที่ 4.8 คำอธิบายยูสเคสไคอะแกรมของการ Check file property

ยูสเคส	Check file property
วัตถุประสงค์	เพื่อทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลบนเว็บเซิร์ฟเวอร์ซึ่งได้แก่ วันที่แก้ไขล่าสุด ขนาดไฟล์ เมื่อถึงเวลาที่กำหนดของเอเจนต์
เงื่อนไขเมื่อเริ่มต้น	Web Fetcher Agent ได้รับสิทธิในการเข้าถึงข้อมูล
แอจेंटที่เกี่ยวข้อง	Timer
รายละเอียด	<ol style="list-style-type: none"> <li>1. Web Fetcher Agent อ่านคุณสมบัติของไฟล์ urllist.xml ที่อยู่ในฝั่งเซิร์ฟเวอร์</li> <li>2. นำคุณสมบัติต่างๆที่อ่านได้จากไฟล์ urllist.xml มาเปรียบเทียบกับข้อมูลเดิม</li> <li>3. หากคุณสมบัติของไฟล์มีการเปลี่ยนแปลง Web Fetcher Agent ทำการบันทึกสถานะการเปลี่ยนแปลงเก็บไว้ และเรียกใช้ยูสเคส Notify Content Change เพื่อเป็นการแจ้งให้ Content Comparing Agent ทราบว่าข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง</li> </ol>

ตารางที่ 4.9 คำอธิบายยูสเคสไคอะแกรมของ Notify Content Change

ยูสเคส	Notify Content Change
วัตถุประสงค์	Web Fetcher Agent ทำการแจ้งให้ Content Comparing Agent ทราบเมื่อมีข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง
แอจेंटที่เกี่ยวข้อง	Content Comparing Agent
สิ่งที่กระตุ้นการทำงาน	ข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง
รายละเอียด	Web Fetcher Agent ส่งชื่อเว็บไซต์ที่มีการเปลี่ยนแปลงไปยัง Content Comparing Agent

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ยูสเคสไดอะแกรม Server Agent

ตารางที่ 4.10 แอ็กเตอร์ที่เกี่ยวข้องกับ Server Agent

แอ็กเตอร์	รายละเอียด
Web Fetcher Agent	เอเจนต์เคลื่อนที่ที่ทำหน้าที่ตรวจสอบการเปลี่ยนแปลงของข้อมูลในหน้าเว็บ
Timer	กำหนดเวลาที่ทำให้ Server Agent ไปบันทึกข้อมูลที่เกี่ยวข้องกับไฟล์บนเว็บเซิร์ฟเวอร์ลงไฟล์ urllist.xml

ตารางที่ 4.11 คำอธิบายยูสเคสไดอะแกรมของการ Record file property to file

ยูสเคส	Record file property to file
วัตถุประสงค์	เพื่อประกาศคุณสมบัติต่างๆของไฟล์ข้อมูล ได้แก่ วันที่แก้ไขล่าสุด ขนาดไฟล์ เมื่อถึงเวลาที่กำหนดของ Server Agent
แอ็กเตอร์ที่เกี่ยวข้อง	Timer
รายละเอียด	<ol style="list-style-type: none"> <li>1. Server Agent ทำการอ่านคุณสมบัติต่างๆ ของไฟล์ข้อมูลในเว็บนั้นๆ ได้แก่ วันที่แก้ไขล่าสุด ขนาดไฟล์ โดยจะทำการอ่านเมื่อถึงเวลาที่กำหนดของเอเจนต์</li> <li>2. Server Agent นำคุณสมบัติต่างๆของไฟล์ ได้แก่ วันที่แก้ไขล่าสุด ขนาดไฟล์ มาบันทึกไว้ในไฟล์ urllist.xml ที่อยู่ในฝั่งเซิร์ฟเวอร์</li> </ol>

ตารางที่ 4.12 คำอธิบายยูสเคสไดอะแกรมของการ Get Access Permission

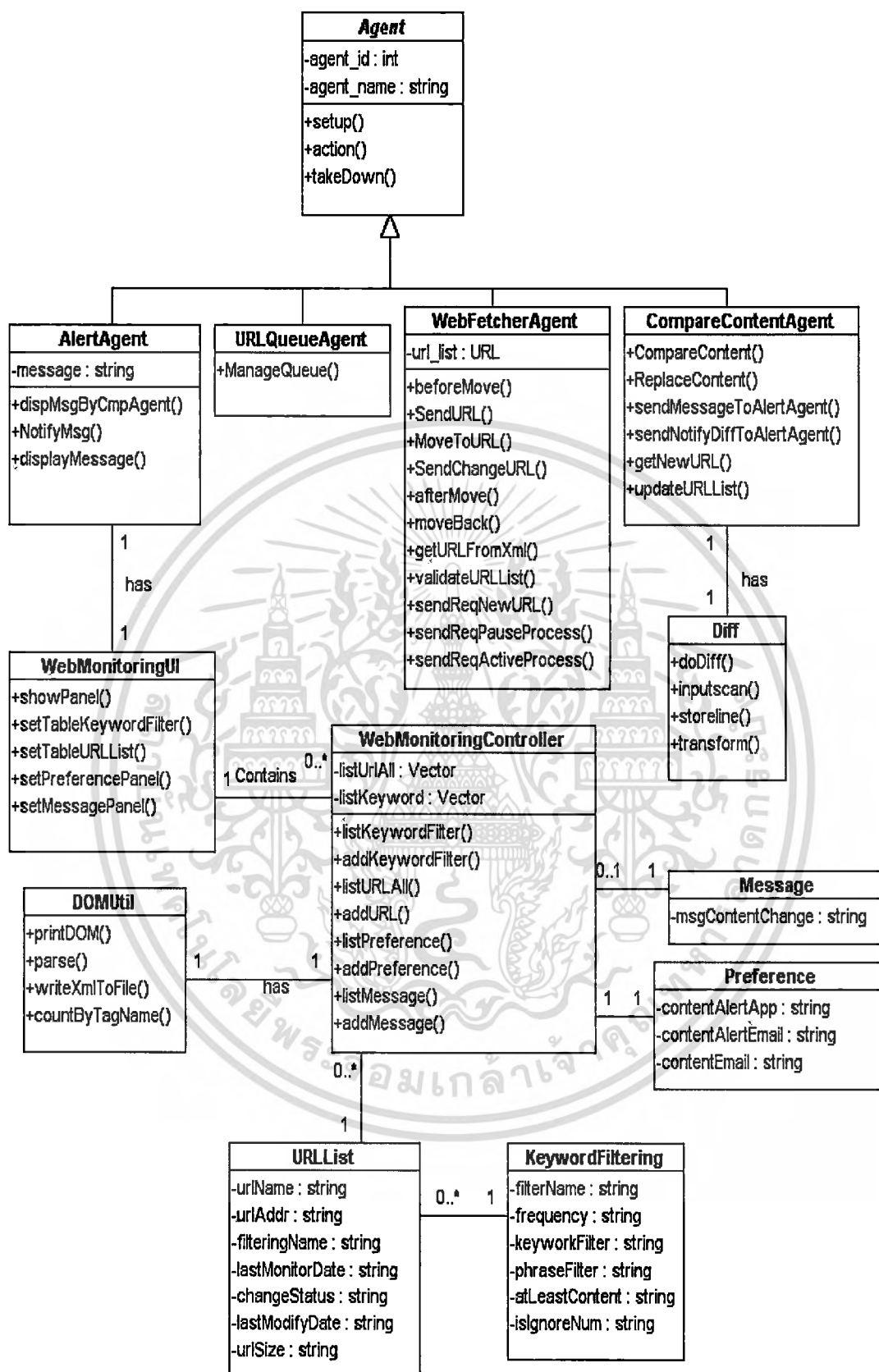
ยูสเคส	Get Access Permission
วัตถุประสงค์	เพื่อทำการตรวจสอบสิทธิในการเข้าถึงข้อมูลบนเซิร์ฟเวอร์ของ Web Fetcher Agent
แอกเตอร์ที่เกี่ยวข้อง	Web Fetcher Agent
สิ่งที่กระตุ้นการทำงาน	เมื่อ Web Fetcher Agent ส่งคำร้องขออนุญาต เพื่อเข้าถึงข้อมูลบนเว็บเซิร์ฟเวอร์
รายละเอียด	<ol style="list-style-type: none"> <li>1. Web Fetcher Agent ส่ง Agent ID ให้ Server Agent</li> <li>2. Server Agent ตรวจสอบสิทธิของ Web Fetcher Agent โดยจะนำ Agent ID มาเข้ารหัสแบบ MD5 และนำมาเปรียบเทียบกับ Agent ID ที่ถูกเข้ารหัสมาจากโคลเอนท์</li> <li>3. หากมีสิทธิ Server Agent จะอนุญาตตามคำร้องขอนั้น Web Fetcher Agent จึงจะสามารถเข้าถึงข้อมูลบนเซิร์ฟเวอร์ได้</li> </ol>

#### 4.3 คลาสไดอะแกรม

คลาสไดอะแกรมของระบบแสดงโครงสร้าง ความสัมพันธ์ระหว่างคลาภายในระบบ รวมถึงคุณสมบัติ และการดำเนินการต่างๆที่สามารถกระทำได้ แสดงดังรูปที่ 4.12 โดยระบบที่ได้ทำการออกแบบจะมีรายละเอียดของแต่ละคลาต่างๆดังนี้

1. คลาส WebMonitoringUI คือคลาสที่ทำหน้าที่เป็นส่วนต่อประสานและใช้ในการติดต่อกับผู้ใช้งาน
2. คลาส Agent คลาสหลักของเอเจนต์ ประกอบด้วยคำสั่ง และคุณสมบัติพื้นฐานในการทำงานของเอเจนต์ภายในเจดแพลตฟอร์ม และเอเจนต์ทุกตัวที่อยู่ภายในระบบจะทำการสืบทอดคุณสมบัติพื้นฐานเหล่านั้นมาใช้ในการทำงานด้วย
3. คลาส URLQueueAgent คือ คลาสที่ทำหน้าที่เป็นเอเจนต์ในการจัดลำดับเว็บไซต์ที่จะให้เอเจนต์เคลื่อนที่ไปเฝ้าสังเกตหรือตรวจสอบให้กับคลาส WebFetcherAgent พร้อมทั้งยังทำหน้าที่ติดต่อกับไฟล์ข้อมูลที่มีรายละเอียดเกี่ยวกับข้อมูลผู้ใช้และข้อมูลของหน้าเว็บที่จะไปเฝ้าสังเกต
4. คลาส WebFetcherAgent คือคลาสที่เป็นเอเจนต์เคลื่อนที่ทำหน้าที่ไปตรวจสอบข้อมูลเบื้องต้นของเว็บไซต์ ได้แก่ วันที่แก้ไขล่าสุดของไฟล์ข้อมูลในหน้าเว็บ และกลับมารายงานผลการเปลี่ยนแปลงต่างๆ ให้กับคลาส CompareContentAgent

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 คลาสไดอะแกรม

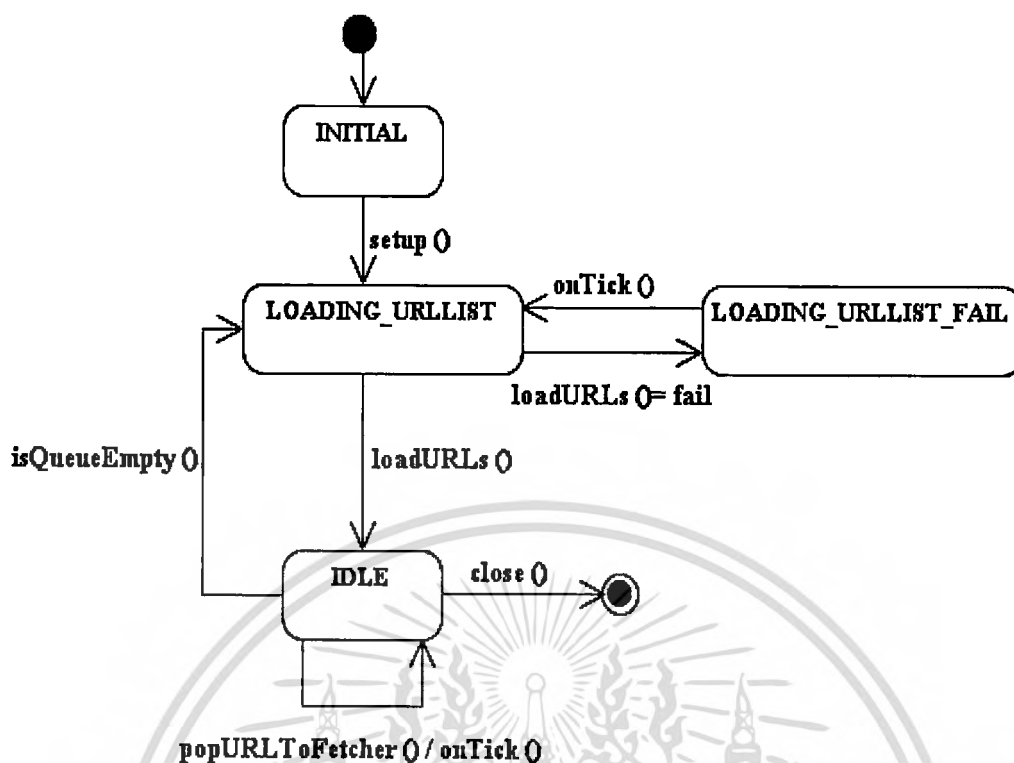
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. คลาส CompareContentAgent และ คลาส Diff คือคลาสที่ทำหน้าที่แยกส่วนที่เป็นเนื้อหาออกจากแท็กคำสั่งใน HTML หากได้รับการรายงานผลจากคลาส WebFetcherAgent ว่าหน้าเว็บนั้นมีการแก้ไข หรือมีการเปลี่ยนแปลงล่าสุด จากนั้นคลาส Diff จะนำข้อมูลใหม่มาเปรียบเทียบกับไฟล์ข้อมูลอันเก่าที่ได้จัดเก็บ พร้อมทั้งทำสัญลักษณ์ส่วนของข้อมูลที่มีความแตกต่างไว้
6. คลาส AlertAgent คือคลาสที่ทำหน้าที่นำข้อมูลที่มีการเปลี่ยนแปลงมาสร้างเป็นรายงานเพื่อแสดงให้ผู้ใช้ทราบ พร้อมทั้งสร้างข้อความที่จะทำการส่งข้อความไปแจ้งผู้ใช้ตามช่องทางที่ผู้ใช้กำหนดไว้
7. คลาส WebMonitoringController และ คลาส DOMUtil คือคลาสเกี่ยวกับการจัดการข้อมูลทั้งหมดในระบบ ทำการจัดเก็บข้อมูลต่างๆลง XML ไฟล์ และทำการเข้าถึงข้อมูลในไฟล์เหล่านั้น
8. คลาส URLList คือเอนทิตีคลาสที่ทำหน้าที่เก็บข้อมูลเกี่ยวกับรายละเอียดของเว็บไซต์ที่ต้องการให้ระบบเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล โดยจะทำการจัดเก็บข้อมูลเหล่านั้นลงไฟล์ urllist.xml
9. คลาส KeywordFiltering คือเอนทิตีคลาสที่ทำหน้าที่เก็บรายละเอียดของตัวคัดกรองที่ผู้ใช้ตั้งค่าไว้สำหรับเป็นตัวคัดกรองข้อมูลที่มีการเปลี่ยนแปลง โดยจะทำการจัดเก็บข้อมูลเหล่านั้นลงไฟล์ filter.xml
10. คลาส Message คือเอนทิตีคลาสที่ทำหน้าที่เก็บรายละเอียดของข้อความเตือนที่ผู้ใช้ตั้งค่าไว้สำหรับให้ระบบนำข้อความนี้ส่งไปแจ้งเตือน เมื่อข้อมูลบนเว็บไซต์มีการเปลี่ยนแปลง โดยจะทำการจัดเก็บข้อมูลนี้ลงไฟล์ alertMessage.xml
11. คลาส Preference คือเอนทิตีคลาสที่ทำหน้าที่เก็บรายละเอียดเกี่ยวกับช่องทางการแจ้งเตือนที่ผู้ใช้กำหนดไว้ เพื่อให้ระบบแจ้งเตือนผ่านช่องทางดังกล่าว โดยจะทำการจัดเก็บข้อมูลนี้ลงไฟล์ preference.xml

#### 4.4 สเตทชาร์ตไดอะแกรม

สเตทชาร์ตไดอะแกรมแสดงถึงลำดับและสถานะการทำงานของเอเจนต์แต่ละตัวภายในระบบซึ่งประกอบไปด้วยเอเจนต์ต่างๆดังนี้

- สเตทชาร์ตไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์จัดลำดับงาน ดังรูปที่ 4.13

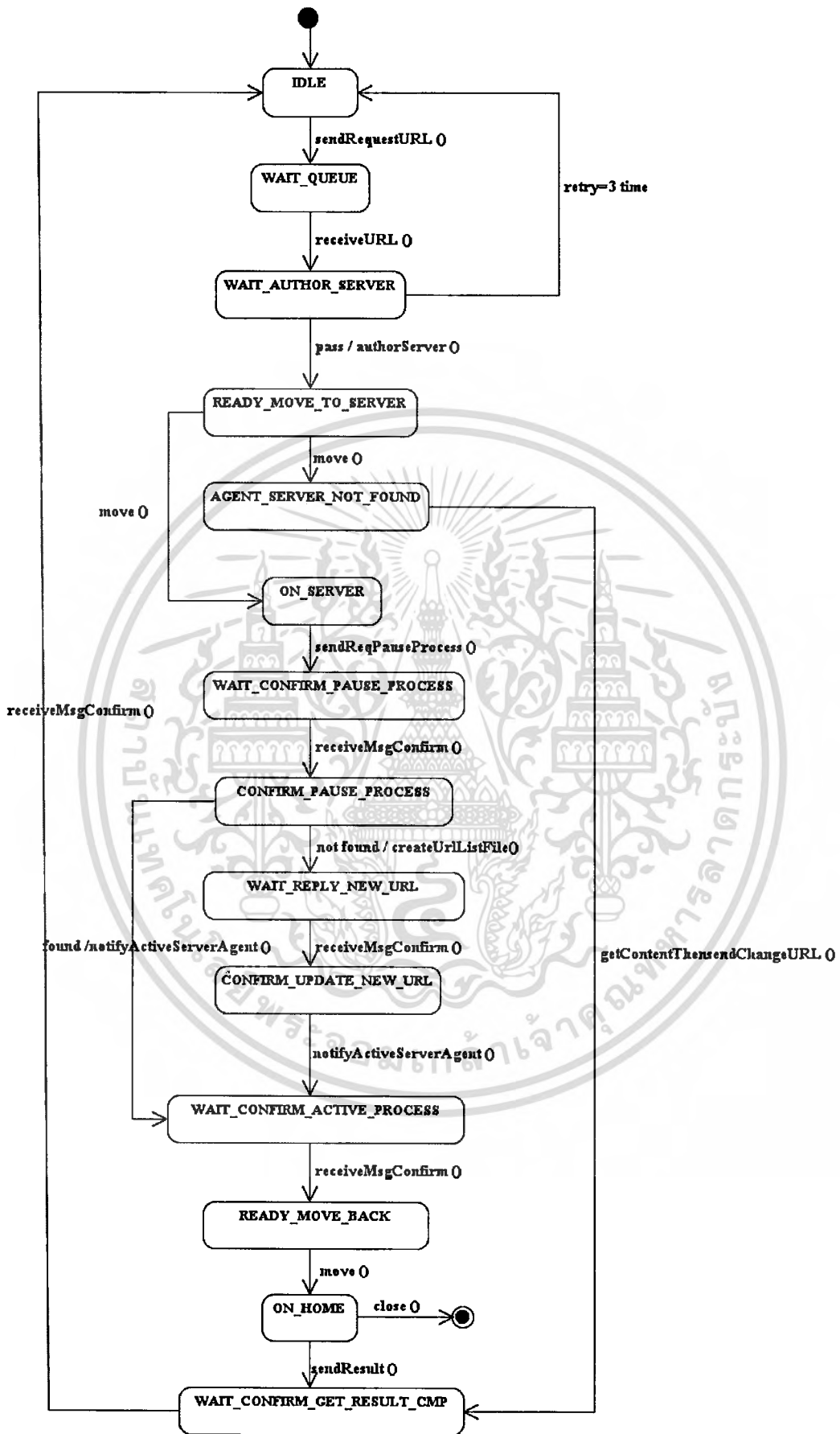


รูปที่ 4.13 สเตตชาร์ทโคโอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์จัดลำดับงาน

จากรูปที่ 4.13 เมื่อเริ่มต้นการทำงานของเอเจนต์จัดลำดับงานจะมีสถานะเป็น INITIAL จากนั้นเมื่อมีการเรียกฟังก์ชัน `setup()` เอเจนต์จัดลำดับงานจะเปลี่ยนสถานะเป็น `LOADING_URLLIST` และเรียกฟังก์ชัน `loadURLs()` เพื่อทำการอ่านข้อมูลรายละเอียดของเว็บไซต์จากไฟล์ `urllist.xml` หากอ่านข้อมูลไม่สำเร็จ เอเจนต์จัดลำดับงานจะเปลี่ยนสถานะเป็น `LOADING_URLLIST_FAIL` และเอเจนต์จัดลำดับงานจะรอให้เกิด `onTick()` เพื่อให้สถานะการทำงานกลับมาสู่สถานะ `LOADING_URLLIST` อีกครั้ง หากการอ่านข้อมูลจากไฟล์สำเร็จ เอเจนต์จัดลำดับงานจะทำการจัดลำดับเว็บไซต์ลงคิว พร้อมกับเปลี่ยนสถานะเป็น `IDLE` เพื่อรอรับคำสั่งจากเอเจนต์เฝ้าสังเกตขอให้มีการส่งรายละเอียดเว็บไซต์ในคิวไปให้ โดยเมื่อได้รับคำสั่งเอเจนต์จัดลำดับงานจะทำการเรียกฟังก์ชัน `popURLToFetcher()` เพื่อนำข้อมูลในคิวส่งให้เอเจนต์เฝ้าสังเกต และหากเอเจนต์จัดลำดับงานอยู่ในสถานะ `IDLE` เอเจนต์จัดลำดับงานจะคอยตรวจสอบข้อมูลในคิว โดยการเรียกฟังก์ชัน `isQueueEmpty()` เพื่อตรวจสอบข้อมูลยังมีอยู่ในคิวหรือไม่ หากข้อมูลในคิวหมด เอเจนต์จัดลำดับงานจะเปลี่ยนสถานะการทำงานเป็น `LOADING_URLLIST`

- สเตตชาร์ทโคโอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เฝ้าสังเกต แสดงดังรูป 4.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



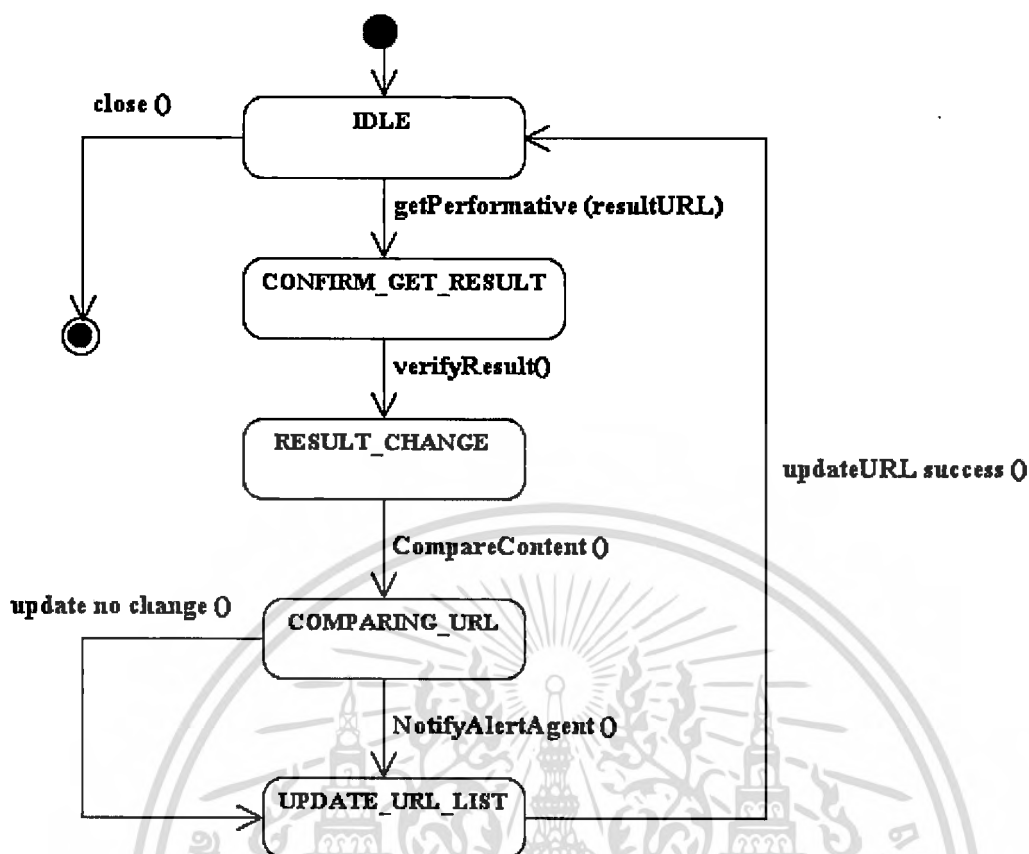
รูปที่ 4.14 สเตตชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เฝ้าสังเกตเอกสารนี้เมื่อเอกสารที่ส่งมอบให้เสร็จสิ้นเพื่อที่เอเจนต์เฝ้าสังเกตจะดำเนินการค้นหาเอกสารใหม่ถ้าหากเอกสารที่ส่งมอบให้หมดอายุหรือถ้าเอกสารที่ส่งมอบให้หมดอายุไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 4.14 เมื่อเริ่มต้นการทำงานของเอเจนต์เฝ้าสังเกตจะมีสถานะเป็น IDLE จากนั้นเอเจนต์เฝ้าสังเกตจะส่งคำสั่ง `sendRequestURL()` ไปยังเอเจนต์จัดลำดับงานเพื่อทำการร้องขอรายละเอียดเว็บไซต์ พร้อมทั้งเปลี่ยนสถานะการทำงานเป็น `WAIT_QUEUE` เพื่อรอข้อมูลจากเอเจนต์จัดลำดับงาน เมื่อได้รับข้อมูลแล้ว เอเจนต์เฝ้าสังเกตจะทำการเปลี่ยนสถานะการทำงานเป็น `WAIT_AUTHOR_SERVER` เพื่อรอตรวจสอบว่าสามารถเคลื่อนที่ไปยังเว็บเซิร์ฟเวอร์นั้นได้หรือไม่ เมื่อได้รับการตอบรับจากเว็บเซิร์ฟเวอร์แล้วจะมีการเปลี่ยนสถานะการทำงานเป็น `READY_MOVE_TO_SERVER` เพื่อเตรียมเคลื่อนที่ไปยังเซิร์ฟเวอร์ จากนั้นจะเคลื่อนที่ไปยังเซิร์ฟเวอร์

ถ้าเอเจนต์เฝ้าสังเกตเคลื่อนที่ไปยังเซิร์ฟเวอร์ไม่สำเร็จ หรือเซิร์ฟเวอร์นั้นไม่รองรับการทำงานของเอเจนต์เคลื่อนที่ สถานะการทำงานจะเปลี่ยนเป็น `AGENT_SERVER_NOT_FOUND` เอเจนต์เฝ้าสังเกตจะทำการส่งข้อความให้เอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์ โดยจะเรียกฟังก์ชัน `getContentThensendChangeURL()` เพื่อให้เอเจนต์เปรียบเทียบไปทำการดึงข้อมูลหรือเนื้อหาของเว็บไซต์มาทำการเปรียบเทียบกับข้อมูลเดิมที่มีอยู่ได้เสีย โดยไม่ต้องรอตรวจสอบผลการเปลี่ยนแปลงจากเอเจนต์เฝ้าสังเกต

ถ้าเอเจนต์เฝ้าสังเกตเคลื่อนที่ไปยังเซิร์ฟเวอร์สำเร็จ สถานะการทำงานจะเปลี่ยนเป็น `ON_SERVER` เอเจนต์เฝ้าสังเกตจะส่งคำร้องขอไปยังเอเจนต์เซิร์ฟเวอร์ เพื่อขออนุญาตเอเจนต์เซิร์ฟเวอร์เข้าถึงข้อมูลบนเว็บเซิร์ฟเวอร์ จากนั้นเอเจนต์เฝ้าสังเกตจะเปลี่ยนสถานะการทำงานเป็น `WAIT_CONFIRM_PAUSE_PROCESS` เมื่อเอเจนต์เฝ้าสังเกตได้รับการอนุญาตให้เข้าถึงข้อมูลบนเซิร์ฟเวอร์แล้ว เอเจนต์เฝ้าสังเกตจะทำการเข้าถึงข้อมูลในไฟล์บนเว็บเซิร์ฟเวอร์ พร้อมทั้งเปลี่ยนสถานะการทำงานเป็น `WAIT_CONFIRM_ACTIVE_PROCESS` และหลังจากอ่านข้อมูลเสร็จเรียบร้อยแล้ว เอเจนต์เฝ้าสังเกตจะส่งข้อความ `receiveMsgConfirm()` แจ้งให้เอเจนต์เซิร์ฟเวอร์ทราบ พร้อมทั้งเปลี่ยนสถานะเป็น `READY_MOVE_BACK` เพื่อเตรียมตัวเคลื่อนที่กลับมายังไคลเอนต์ เมื่อเอเจนต์เฝ้าสังเกตเคลื่อนที่กลับมาถึงไคลเอนต์แล้ว จะเปลี่ยนสถานะการทำงานเป็น `ON_HOME` หลังจากนั้นเอเจนต์เฝ้าสังเกตจะส่งข้อความ `sendResult()` แจ้งเอเจนต์เปรียบเทียบทราบและเพื่อส่งข้อมูลของเว็บไซต์ที่มีการเปลี่ยนแปลงให้ พร้อมทั้งเปลี่ยนสถานะการทำงานเป็น `WAIT_CONFIRM_GET_RESULT` เพื่อรอให้เอเจนต์เปรียบเทียบตอบกลับและส่งผลลัพธ์ให้เอเจนต์เปรียบเทียบทำงานต่อไป

- สดตทซาร์ทโคอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์ แสดงดังรูป 4.15

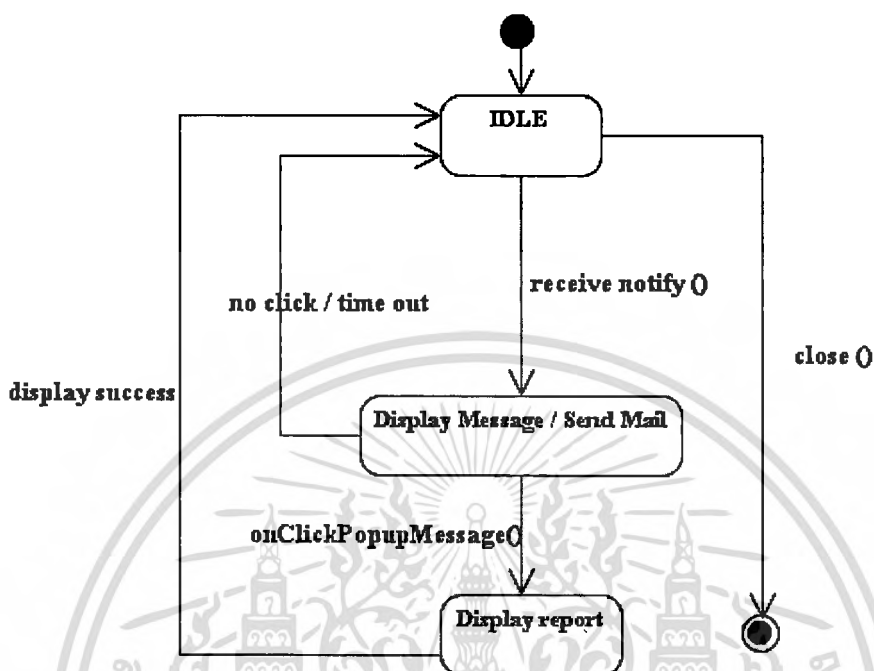


รูปที่ 4.15 สเตตชาร์ทโคดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์

จากรูป 4.15 เมื่อเริ่มต้นการทำงานของเอเจนต์เปรียบเทียบจะมีสถานะเป็น IDLE เมื่อได้รับข้อความจากเอเจนต์เฝ้าสังเกต เอเจนต์เปรียบเทียบจะทำการ `getPerformative()` และเปลี่ยนสถานะการทำงานเป็น `CONFIRM_GET_RESULT` จากนั้นเอเจนต์เปรียบเทียบจะทำการตรวจสอบข้อมูลที่ได้รับมาจากเอเจนต์เฝ้าสังเกต พร้อมทั้งเปลี่ยนสถานะการทำงานเป็น `RESULT_CHANGE` จากนั้นเอเจนต์เปรียบเทียบจะเริ่มดำเนินการดึงไฟล์ข้อมูลเก่าขึ้นมาเพื่อรอทำการเปรียบเทียบ และเปลี่ยนสถานะการทำงานเป็น `COMPARING_URL` พร้อมทั้งทำการดำเนินการเปรียบเทียบเนื้อหาของเว็บไซต์ หากพบว่าข้อมูลมีการเปลี่ยนแปลง เอเจนต์เปรียบเทียบจะส่งข้อความแจ้งให้กับเอเจนต์แจ้งเตือนทราบ จากนั้นจะทำการเปลี่ยนสถานะการทำงานเป็น `UPDATE_URL_LIST` พร้อมทั้งทำการปรับปรุงรายละเอียดและสถานะของเว็บไซต์ในไฟล์ `urllist.xml` และหากไม่พบว่าข้อมูลมีการเปลี่ยนแปลง เอเจนต์เปรียบเทียบจะทำการเปลี่ยนสถานะการทำงานเป็น `UPDATE_URL_LIST` พร้อมทั้งทำการปรับปรุงรายละเอียดและสถานะของเว็บไซต์ในไฟล์ `urllist.xml` เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์แจ้งเตือน แสดงดังรูป 4.16



รูปที่ 4.16 สเตทชาร์ทไดอะแกรมแสดงการเปลี่ยนสถานะการทำงานของเอเจนต์แจ้งเตือน

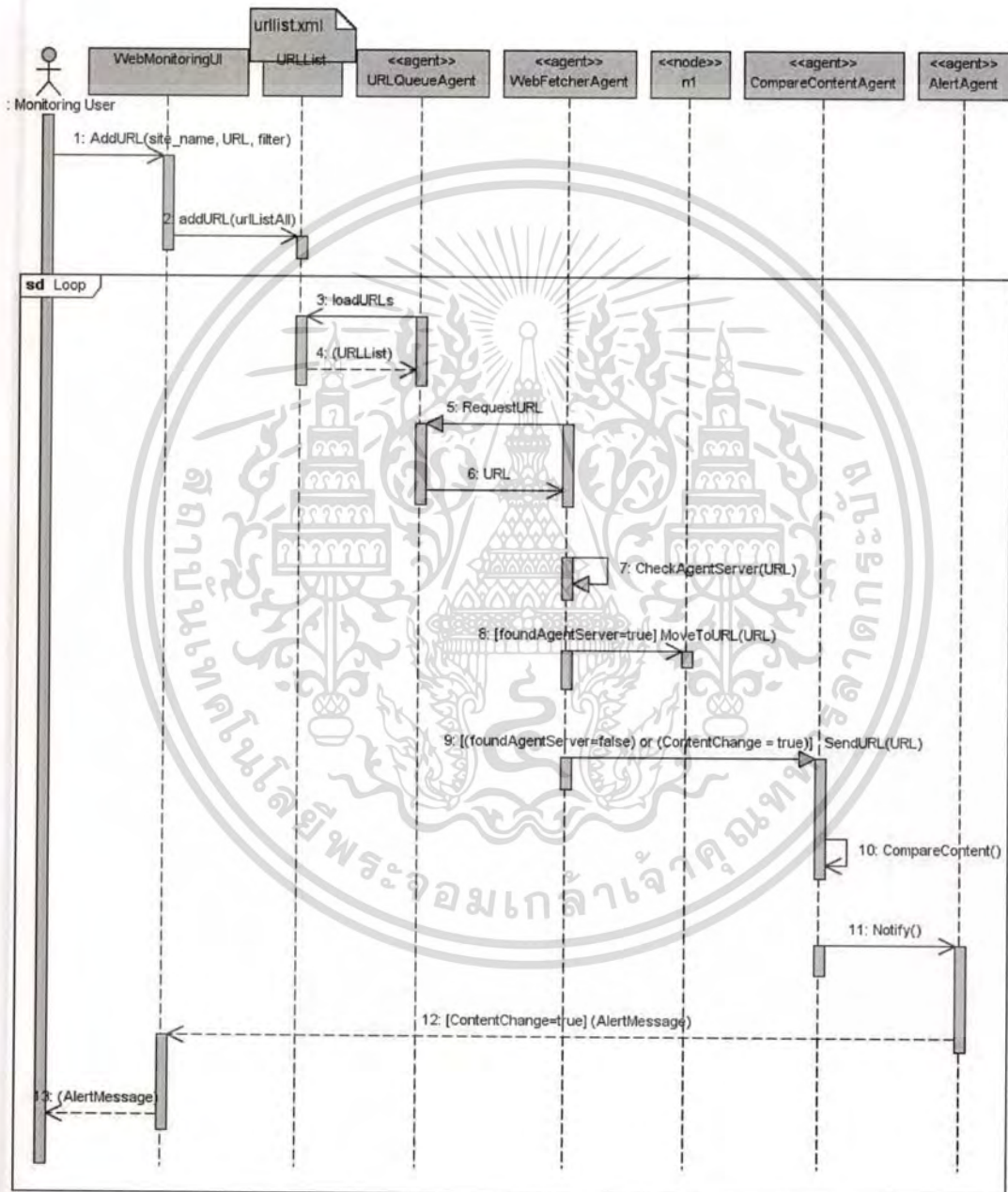
จากรูป 4.16 เมื่อเริ่มต้นการทำงานของเอเจนต์แจ้งเตือนมีสถานะการทำงานเป็น IDLE เมื่อเอเจนต์แจ้งเตือน ได้รับข้อความจากเอเจนต์เปรียบเทียบ เอเจนต์แจ้งเตือนจะส่งข้อความแจ้งเตือนผ่านแอปพลิเคชันหรือแจ้งเตือนผ่านอีเมล หากผู้ใช้งานทำการคลิกที่ป๊อปอัพเอเจนต์แจ้งเตือนจะแสดงรายงานการเปลี่ยนข้อมูลของเว็บไซต์ผ่านแอปพลิเคชันให้ผู้ใช้เห็น เมื่อแสดงผลเรียบร้อยแล้วเอเจนต์แจ้งเตือนจะกลับไปสู่สถานะ IDLE

#### 4.5 ซีเควนซ์ไดอะแกรม

แผนภาพจำลองกระบวนการทำงานที่เป็นกิจกรรมของระบบซึ่งกิจกรรมหนึ่งๆ นั้นเกิดจากการที่อ็อบเจกต์หนึ่งได้ตอบกับอีกอ็อบเจกต์หนึ่งด้วยการรับส่งข้อความกัน สำหรับในระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ จะแสดงการรับส่งข้อความกันระหว่างเอเจนต์แต่ละตัวที่อยู่ภายในระบบและการแสดงการโต้ตอบระหว่างเอเจนต์กับผู้ใช้ด้วย สำหรับซีเควนซ์ไดอะแกรมของระบบประกอบไปด้วย 2 ส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

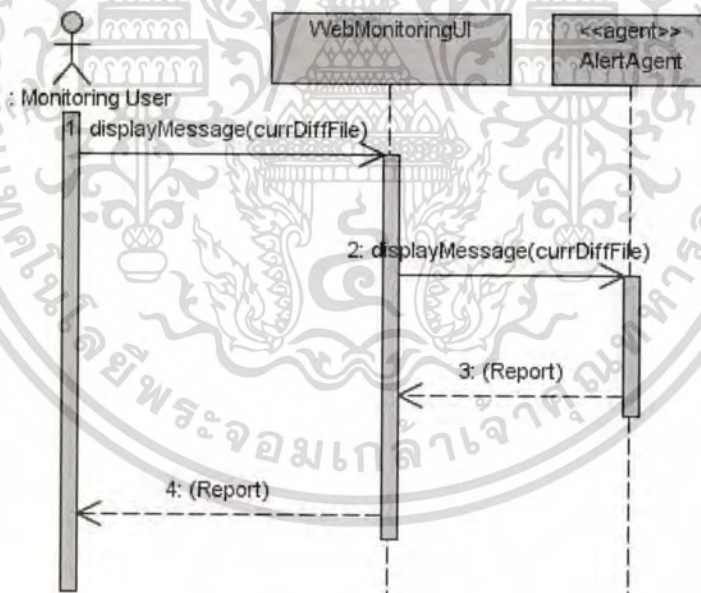
1. ซีควেনซ์ไดอะแกรมของการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลข่าวสารที่หน้าเว็บตามที่ผู้ใช้กำหนด ระบบจะทำการเฝ้าสังเกตหน้าเว็บ เมื่อหน้าเว็บมีการเปลี่ยนแปลงข้อมูล ระบบจะทำการส่งข้อความแจ้งให้กับผู้ใช้ทราบตามช่องทางที่ผู้ใช้กำหนด ดังรูปที่ 4.17
2. ซีควেনซ์ไดอะแกรมของการเรียกดูรายงานการเปลี่ยนแปลงของหน้าเว็บ ซึ่งรายงานเป็นรายงานที่เกิดจากการเปลี่ยนแปลงล่าสุดที่ระบบได้ตรวจสอบมา ดังรูปที่ 4.18



รูปที่ 4.17 ซีควেনซ์ไดอะแกรมของการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.17 แสดงการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนเว็บไซต์ โดยเริ่มจากผู้ใช้งานเพิ่มรายละเอียดเว็บไซต์กับส่วนต่อประสานกับผู้ใช้ ส่วนต่อประสานกับผู้ใช้จะทำการเพิ่มรายละเอียดเหล่านั้นลงในไฟล์ `urllist.xml` เมื่อถึงเวลาที่กำหนด `URLQueueAgent` จะทำการอ่านข้อมูลจากไฟล์ `urllist.xml` มาเรียงลำดับ จากนั้น `WebFetcherAgent` จะส่งคำร้องขอไปยัง `URLQueueAgent` เพื่อขอรายละเอียดเว็บไซต์ที่จะเดินทางไปทำการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล เมื่อได้รับรายชื่อเว็บไซต์แล้ว `WebFetcherAgent` จะทำการตรวจสอบว่าเว็บไซต์เวอร์ชันที่รองรับการทำงานของเขาเจนท์เคลื่อนที่หรือไม่ จากนั้นจะเคลื่อนที่ไปยังเว็บไซต์เวอร์ชันที่รองรับการทำงานของเขาเจนท์เคลื่อนที่ และทำการเปรียบเทียบข้อมูลบนเว็บไซต์เวอร์ชันว่ามีการเปลี่ยนแปลงหรือไม่ หากข้อมูลมีการเปลี่ยนแปลง `WebFetcherAgent` จะส่งข้อความ `SendURL ()` ให้กับ `CompareContentAgent` เพื่อให้ `CompareContentAgent` ทำการดึงเนื้อหาของเว็บไซต์นั้นมาทำการเปรียบเทียบกับข้อมูลเดิมที่มีอยู่ เพื่อตรวจสอบว่าข้อมูลมีการเปลี่ยนแปลงหรือไม่ หากพบการเปลี่ยนแปลงของข้อมูลตามตัวคัดกรองที่ผู้ใช้กำหนด `CompareContentAgent` จะทำการส่งข้อความ `Notify ()` แจ้งให้ `AlertAgent` ทราบ และ `AlertAgent` จะทำการส่งข้อความ `AlertMessage()` เพื่อแจ้งผลการเปลี่ยนแปลงแก่ผู้ใช้



รูปที่ 4.18 ซีควเอนซ์ไดอะแกรมของการเรียกดูรายงาน

จากรูปที่ 4.18 แสดงการเรียกดูรายงานผลการเปลี่ยนแปลงของข้อมูลบนเว็บไซต์ โดยเริ่มจากผู้ใช้คลิกที่หน้าจอป๊อปอัพที่มีการแจ้งเตือนว่าเกิดการเปลี่ยนแปลงข้อมูลบนเว็บไซต์ ส่วนต่อประสานกับผู้ใช้จะส่งข้อความ `displayMessage` ไปยัง `AlertAgent` จากนั้น `AlertAgent` จะส่งไฟล์รายงานผลมาแสดงยังส่วนต่อประสานกับผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.6 การออกแบบการจับเก็บข้อมูล

ระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บจะออกแบบการเก็บข้อมูลในรูปแบบของไฟล์ที่บันทึกในเครื่องของผู้ใช้ โดยจะประกอบไปด้วยไฟล์ 5 ประเภทคือ

1. `urllist.xml` คือไฟล์ที่เก็บรายละเอียดของเว็บไซต์ที่ผู้ใช้ระบุ ได้แก่ ชื่อเว็บไซต์ วันที่แก้ไขล่าสุด วันที่เวลาที่เอเจนท์ไปทำการตรวจสอบล่าสุด สถานะการตรวจสอบ การตั้งค่าการตรวจสอบการเปลี่ยนแปลง ขนาดไฟล์ เป็นต้น
2. `filter.xml` คือไฟล์ที่เก็บรายละเอียดการตั้งค่าการตรวจสอบการเปลี่ยนแปลง หรือคำคัดกรองตามที่ผู้ใช้กำหนด
3. `preference.xml` คือไฟล์ที่เก็บรายละเอียดการตั้งค่าช่องทางการแจ้งเตือน
4. `alertMessage.xml` คือ ไฟล์ที่เก็บรายละเอียดการตั้งค่าข้อความในการแจ้งเตือน
5. `old_url_content.txt` คือ ไฟล์ที่เก็บเนื้อหาเก่าของเว็บไซต์
6. `new_url_content.txt` คือ ไฟล์ที่เก็บเนื้อหาล่าสุดของเว็บไซต์
7. `urlname_date_time.html` คือไฟล์ที่แสดงรายงานผลการเปลี่ยนแปลงของข้อมูล

รายละเอียดข้อมูลที่เก็บในไฟล์แต่ละประเภทจะเก็บเป็นแถวที่เรียงลำดับฟิลด์ที่กำหนด โดยต้นบรรทัดของแต่ละแถวจะเก็บชื่อฟิลด์ เพื่อให้ง่ายต่อการนำข้อมูลมาใช้ ซึ่งรายละเอียดแสดงดังตารางที่ 4.13 ถึง 4.16 ตามลำดับ

ตารางที่ 4.13 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ `urllist.xml`

ชื่อ	ชนิดข้อมูล	คำอธิบาย
URLName	String	ชื่อเว็บไซต์
URLAddr	String	ที่อยู่เว็บไซต์
URLFilter	String	ชื่อการตั้งค่าการตรวจสอบการเปลี่ยนแปลง อ้างอิงไปที่ไฟล์ <code>filter.xml</code> - FilterName
LastMonitorDate	String	วันที่เวลาที่เอเจนท์ไปตรวจสอบล่าสุด
ChangeStatus	String	สถานะการตรวจสอบ S = เว็บไซต์ใหม่ C = ข้อมูลมีการเปลี่ยนแปลง N = ข้อมูลไม่มีการเปลี่ยนแปลง
LastModifyDate	String	วันที่แก้ไขเว็บไซต์ล่าสุด
ContentFileSize	Number	ขนาดของไฟล์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตัวอย่างข้อมูลที่เก็บในไฟล์ urlist.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<URLList>
  <URL>
    <URLName>Posttoday</URLName>
    <URLAddr>www.posttoday.com</URLAddr>
    <URLFilter>การเงิน</URLFilter>
    <LastMonitorDate>2553.03.23 พ.ศ. at 10:58:40
GMT+07:00</LastMonitorDate>
    <ChangeStatus>N</ChangeStatus>
    <LastModifyDate>2553.03.23 พ.ศ. at 10:47:01
GMT+07:00</LastModifyDate>
    <ContentFileSize>11584 Bytes</ContentFileSize>
  </URL>
</URLList>
```

## ตารางที่ 4.14 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ filter.xml

ชื่อ	ชนิดข้อมูล	คำอธิบาย
FilterName	String	ชื่อการตั้งค่าการตรวจสอบการเปลี่ยนแปลง
Frequency	String	ความถี่ในการให้เงื่อนไขไปทำการเฝ้าสังเกต
Keyword	String	ตั้งคำคำหลัก
Phrase	String	ตั้งคำกลุ่มคำหลัก หรือวลี
AtLeastContent	Number	ตั้งค่าติดตามปริมาณการเปลี่ยนแปลงเมื่อเทียบกับเนื้อหาทั้งหมดในหน้าเว็บไซต์ (%)
IgnoreNumber	String	ตั้งค่าการตรวจสอบข้อมูลที่เป็นตัวเลข Y = ข้อมูลที่เปลี่ยนแปลงมีเฉพาะตัวเลข ระบบไม่ต้องแจ้งเตือน N = ระบบแจ้งเตือนปกติ ไม่สนใจชนิดของข้อมูล

## ตัวอย่างข้อมูลที่เก็บในไฟล์ filter.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Filters>
  <Filter>
    <FilterName>การเงิน</FilterName>
    <Frequency>5 Minutes</Frequency>
    <Keyword>หุ้น ตราสารหนี้ กองทุน</Keyword>
    <Phrase>อัตราดอกเบี้ยเงินฝาก ธนาคารกรุงเทพ</Phrase>
    <AtLeastContent>0</AtLeastContent>
    <IgnoreNumber>false</IgnoreNumber>
  </Filter>
</Filters>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.15 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ preference.xml

ชื่อ	ชนิดข้อมูล	คำอธิบาย
ContentAlertApp	String	ตั้งค่าแจ้งเตือนเมื่อมีข้อมูลเปลี่ยนแปลง True = แจ้งเตือนผ่านแอปพลิเคชัน False = ไม่แจ้งเตือนผ่านแอปพลิเคชัน
ContentAlertEmail	String	ตั้งค่าแจ้งเตือนเมื่อมีข้อมูลเปลี่ยนแปลง True = แจ้งเตือนผ่านอีเมล False = ไม่แจ้งเตือนผ่านอีเมล
ContentEmail	String	อีเมลที่ต้องการให้ระบบส่งข้อความเตือน

ตัวอย่างข้อมูลที่เก็บในไฟล์ preference.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Preference>
  <ContentAlertApp>true</ContentAlertApp>
  <ContentAlertEmail>true</ContentAlertEmail>
  <ContentEmail>maliwan_lao@hotmail.com</ContentEmail>
</Preference>
```

ตารางที่ 4.16 คำอธิบายรายละเอียดข้อมูลที่เก็บในไฟล์ alertMessage.xml

ชื่อ	ชนิดข้อมูล	คำอธิบาย
MsgContentChange	String	ข้อความเตือนเมื่อข้อมูลมีการเปลี่ยนแปลง

ตัวอย่างข้อมูลที่เก็บในไฟล์ alertMessage.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<Message>
  <MsgContentChange>Content Change</MsgContentChange>
</Message>
```

## บทที่ 5

### การพัฒนาระบบ

#### 5.1 เครื่องมือและภาษาที่ใช้ในการพัฒนา

การพัฒนาระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ในโครงการนี้ใช้เครื่องมือและภาษาในการพัฒนาโปรแกรมดังนี้

##### 5.1.1 ฮาร์ดแวร์

เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนาและทดสอบระบบ มีคุณสมบัติดังนี้

- เครื่องคอมพิวเตอร์ส่วนบุคคล Fujitsu S6520
- ซีพียู Core 2 Duo Processor 2.8GHz
- หน่วยความจำหลักขนาด 4 กิกะไบต์
- ฮาร์ดดิสก์ ความจุ 500 กิกะไบต์
- หน่วยแสดงผล S3 Graphics Chrome 430 ULP หน่วยความจำ ขนาด 256 MB

##### 5.1.2 ซอฟต์แวร์

ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ มีดังนี้

- ระบบปฏิบัติการ Microsoft Windows XP Professional Service Pack 3
- Java Standard Edition (J2SE) 5.0 ใช้เป็นเฟรมเวิร์คในการพัฒนาแอปพลิเคชันด้วยภาษาจาวา
- Java Agent Development Framework (JADE) 3.7 ใช้เป็นแพลตฟอร์มในการทำงานของเอเจนต์

##### 5.1.3 เครื่องมือ

เครื่องมือที่ใช้ในการพัฒนาระบบ มีดังนี้

- NetBeans IDE 6.7.1 ใช้เป็นเครื่องมือในการเขียนโปรแกรมด้วยภาษาจาวา
- Visual Paradigm for UML 6.3 Enterprise Edition ใช้เป็นเครื่องมือในการเขียน

แผนภาพของไดอะแกรมต่างๆในช่วงการวิเคราะห์และออกแบบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.2 การจำลองเอเจนต์เซิร์ฟเวอร์

การจำลองเอเจนต์เซิร์ฟเวอร์ เพื่อจำลองการทำงานของเอเจนต์ฝั่งเซิร์ฟเวอร์ที่คอยทำหน้าที่เก็บรายละเอียดเบื้องต้นของเว็บไซต์ในเซิร์ฟเวอร์เหล่านั้น ซึ่งรายละเอียดที่เก็บจะประกอบไปด้วยวันที่มีการแก้ไขข้อมูลบนเว็บไซต์ล่าสุด และขนาดไฟล์ของเว็บไซต์นั้นๆ เอเจนต์เซิร์ฟเวอร์จะทำการเก็บข้อมูลเหล่านี้อย่างอัตโนมัติ และทำงานเป็นระยะเวลาตามที่ระบบกำหนด นอกจากนี้ เอเจนต์เซิร์ฟเวอร์ยังทำหน้าที่ตรวจสอบความปลอดภัยของเอเจนต์เคลื่อนที่ ที่มาจากโคลนที่เข้ามาขอเข้าถึงข้อมูลบนเซิร์ฟเวอร์ ว่าเป็นเอเจนต์เคลื่อนที่เหล่านั้นของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์จริงหรือไม่

การรันเอเจนต์เซิร์ฟเวอร์ จะต้องทำการตั้งค่าพารามิเตอร์สำหรับสร้างคอนเทนเนอร์ให้กับเอเจนต์เซิร์ฟเวอร์อยู่บนเจดแพลตฟอร์ม และเริ่มต้นทำการรันเจดแพลตฟอร์มขึ้นมาดังนี้

```
>> java jade.Boot -gui
```

จากคำสั่งข้างต้นจะทำให้สามารถรันเจดแพลตฟอร์ม โดยภายในแพลตฟอร์มจะประกอบไปด้วยบริการพื้นฐานต่างๆเริ่มต้นขึ้น และคอนเทนเนอร์หลักที่ทำหน้าที่จัดการควบคุมคอนเทนเนอร์อื่นๆบนเจดแพลตฟอร์มจะเกิดขึ้นด้วย

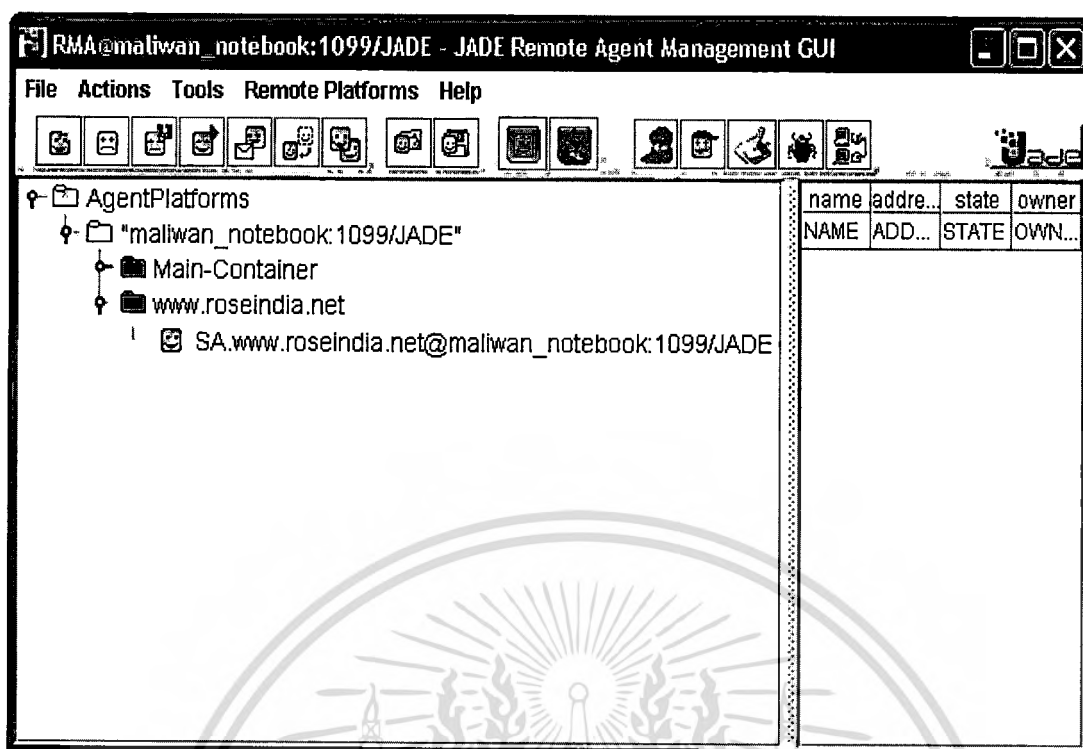
สำหรับออปชัน `-gui` จะทำให้ส่วนต่อประสานของเจดแพลตฟอร์มเกิดขึ้น ซึ่งจะช่วยให้ผู้พัฒนาระบบสามารถตรวจสอบและติดตามการทำงานของเอเจนต์ต่างๆบนแพลตฟอร์มได้

```
>> java jade.Boot -container -container-name www.roseindia.net -host 192.2.200.143 SA.www.roseindia.net:monitoring.server.ServerAgent
```

จากนั้นจะทำการรันเอเจนต์เซิร์ฟเวอร์ขึ้นมา โดยจะกำหนดออปชันต่างๆดังนี้

- `container` สำหรับสร้างคอนเทนเนอร์ เพื่อเก็บเอเจนต์ไว้ในเจดแพลตฟอร์ม
- `container-name` สำหรับตั้งชื่อคอนเทนเนอร์
- `host` กำหนดว่าจะให้เอเจนต์นี้ไปรันที่โฮสต์ตามที่ระบุ

เมื่อรันเอเจนต์เซิร์ฟเวอร์สำเร็จ ภายในเจดแพลตฟอร์มจะปรากฏคอนเทนเนอร์ของเอเจนต์เซิร์ฟเวอร์ ดังรูปที่ 5.1



รูปที่ 5.1 ส่วนต่อประสานของเจดแพลตฟอร์ม

### 5.3 การพัฒนาระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์

การพัฒนาระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บด้วยมัลติเอเจนต์ เป็นการกล่าวถึงรายละเอียดขั้นตอนการทำงานของเอเจนต์แต่ละตัวภายในระบบ ดังต่อไปนี้

#### 5.3.1 เอเจนต์จัดลำดับงาน

เอเจนต์จัดลำดับงาน หรือคลาส QueueAgent เป็นเอเจนต์ที่ทำหน้าที่อ่านรายชื่อเว็บไซต์จากไฟล์ข้อมูล และนำมาทำการจัดลำดับเพื่อให้ระบบทำการเฝ้าสังเกตตามลำดับก่อนหลัง พร้อมทั้งทำการส่งรายละเอียดต่างๆของเว็บไซต์ให้กับเอเจนต์เฝ้าสังเกตไปทำงานต่อไป

หลักการทำงานเริ่มจากเอเจนต์จัดลำดับงานจะทำการตั้งค่าสถานะการทำงานเป็น LOADING\_URLLIST จากนั้นตรวจสอบว่ามีไฟล์ urlist.xml อยู่ในระบบจริงหรือไม่ ถ้ามีไฟล์อยู่ในระบบจริง เอเจนต์จัดลำดับงานจะทำการอ่านข้อมูลรายชื่อเว็บไซต์มาจากไฟล์ urlist.xml แล้วนำมาทำการตรวจสอบและเรียงลำดับด้วยการนำวันเวลาของการตรวจสอบล่าสุดมาเรียงลำดับจากน้อยไปมาก แล้วจึงนำรายชื่อเว็บไซต์เหล่านั้นไปเก็บไว้ในตัวแปรอาร์เรย์ เพื่อรอการส่งต่อให้กับเอเจนต์เฝ้าสังเกต

### 5.3.2 เอเจนต์เฝ้าสังเกต

เอเจนต์เฝ้าสังเกต หรือคลาส WebFetcherAgent เป็นเอเจนต์เคลื่อนที่ที่ทำหน้าที่เดินทางไปยังเว็บไซต์ต่างๆตามที่เอเจนต์จัดลำดับงานแจ้ง และทำการเฝ้าสังเกตการเปลี่ยนแปลงของข้อมูลที่หน้าเว็บไซต์จากไฟล์ข้อมูลที่เอเจนต์เซิร์ฟเวอร์ได้ทำการประกาศไว้ หากข้อมูลเบื้องต้นได้แก่ วันที่แก้ไขไฟล์เว็บไซต์ล่าสุด และขนาดไฟล์เว็บไซต์ล่าสุด ถูกตรวจพบว่ามี การเปลี่ยนแปลง เอเจนต์เฝ้าสังเกตจะทำการเคลื่อนที่กลับมาที่โคลเอเจนต์ และทำการแจ้งให้เอเจนต์เปรียบเทียบทราบและดำเนินการต่อไป

### 5.3.3 เอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์

เอเจนต์เปรียบเทียบเนื้อหาเว็บไซต์ หรือคลาส CompareContentAgent เป็นเอเจนต์ที่ทำหน้าที่ในการดึงข้อมูลจากหน้าเว็บไซต์มาทำการแยกส่วนที่เป็นเนื้อหาออกมาจากแท็กคำสั่ง และทำหน้าที่เปรียบเทียบข้อมูลใหม่ที่ได้กับข้อมูลก่อนหน้า หากข้อมูลมีการเปลี่ยนแปลงตัวเอเจนต์นี้จะทำการแจ้งให้เอเจนต์แจ้งเตือนทราบและดำเนินการต่อไป

ลักษณะการเปรียบเทียบไฟล์ข้อมูลเก่ากับไฟล์ข้อมูลใหม่ โดยเอเจนต์จะอ่านข้อมูลมาทีละบรรทัด แล้วนำมาเก็บไว้ในโหนดที่ทำการสร้างเป็นลักษณะของไบนารีทรีดังนี้ (Darwin. 2004)

```
void inputscan(fileInfo pinfo) throws IOException {
    String linebuffer;
    pinfo.maxLine = 0;
    if (pinfo.fileBuff != null) { //if use charset
        while ((linebuffer = pinfo.fileBuff.readLine()) != null)
        {
            storeline(linebuffer, pinfo);
        }
    } else {
        while ((linebuffer = pinfo.file.readLine()) != null) {
            storeline(linebuffer, pinfo);
        }
    }
}
```

```
void storeline(String linebuffer, fileInfo pinfo) {
    int linenum = ++pinfo.maxLine;
    if (linenum > fileInfo.MAXLINECOUNT) {
        System.err.println("MAXLINECOUNT exceeded, must stop.");
        System.exit(1);
    }
    pinfo.symbol[linenum] = node.addSymbol(linebuffer, pinfo ==
oldinfo, linenum, ps);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

static node addSymbol(String pline, boolean inoldfile, int linenum,
PrintStream ps) {
    node pnode;
    pnode = matchsymbol(pline, ps);
    if (pnode.linestate == freshnode) {
        pnode.linestate = inoldfile ? oldonce : newonce;
    } else {
        if ((pnode.linestate == oldonce && !inoldfile) ||
            (pnode.linestate == newonce && inoldfile)) {
            pnode.linestate = bothonce;
        } else {
            pnode.linestate = other;
        }
    }
    if (inoldfile) {
        pnode.linenum = linenum;
    }
    return pnode;
}

```

จากนั้นระบบจะนำข้อมูลนั้นมาทำการเปรียบเทียบกับข้อมูลภายในทรี เพื่อตรวจสอบว่าข้อมูลมีการซ้ำกับข้อมูลในตำแหน่งใดของทรีแต่ละชุดหรือไม่ และทำการกำหนดสถานะให้กับข้อมูลแต่ละบรรทัดนั้น

```

void scanunique() {
    int oldline, newline;
    node psymbol;
    for (newline = 1; newline <= newinfo.maxLine; newline++) {
        psymbol = newinfo.symbol[newline];
        if (psymbol.symbolIsUnique()) {
            oldline = psymbol.linenum;
            newinfo.other[newline] = oldline;
            oldinfo.other[oldline] = newline;
        }
    }
    newinfo.other[0]=0;
    oldinfo.other[0]=0;
    newinfo.other[newinfo.maxLine + 1]=oldinfo.maxLine + 1;
    oldinfo.other[oldinfo.maxLine + 1]=newinfo.maxLine + 1;
}

```

### 5.3.4 เอนเจนท์แจ้งเตือน

เอนเจนท์แจ้งเตือน หรือคลาส AlertAgent ทำหน้าที่รองรับผลการเปลี่ยนแปลงที่ได้จากการเปรียบเทียบของเอนเจนท์เปรียบเทียบเนื้อหาเว็บไซต์ โดยจะทำการแจ้งเตือนให้ผู้ใช้งานทราบผ่านส่วนต่อประสานกับผู้ใช้ หรือทำการแจ้งเตือนผ่านอีเมลตามที่ผู้ใช้กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการแจ้งเตือนผ่านอีเมล ระบบจะมีการตั้งค่าของอีเมลผู้ส่งของระบบไว้ในไฟล์ properties.xml ดังนี้

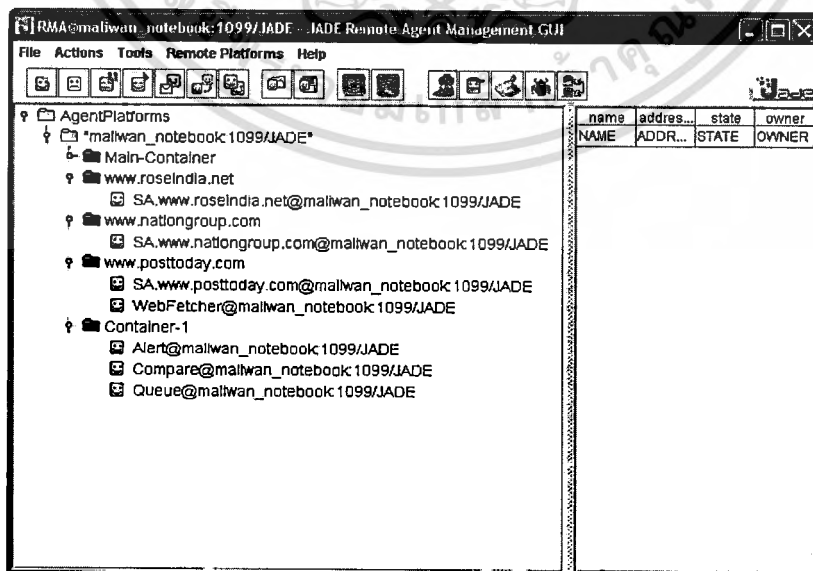
```
<entry key="mail.smtp.host">smtp.gmail.com</entry>
<entry key="mail.smtp.port">587</entry>
<entry key="mail.smtp.auth">true</entry>
<entry key="mail.smtp.starttls.enable">true</entry>
<entry key="mail.auth.user">WebContentMonitoring@gmail.com</entry>
<entry key="mail.auth.pwd">0123456789WebContentMonitoring</entry>
<entry key="mail.sender">WebContentMonitoring@gmail.com</entry>
```

### 5.3.5 การรันระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์

สำหรับการรันระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์ จะต้องทำการตั้งค่าพารามิเตอร์ สำหรับสร้างคอนเทนเนอร์ให้กับเอเจนต์ต่างๆอยู่บนเจดแพลตฟอร์มดังนี้

```
>> java jade.Boot -container -host 192.2.200.143
Alert:monitoring.alert.AlertAgent
Compare:monitoring.compare.CompareContentAgent
Queue:monitoring.queue.QueueAgent
WebFetcher:monitoring.webfetcher.WebFetcherAgent
```

เมื่อรันระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์สำเร็จ ภายในเจดแพลตฟอร์มจะปรากฏคอนเทนเนอร์ของเอเจนต์ ดังรูปที่ 5.2 โดยเอเจนต์ทุกตัวของระบบจะถูกจัดเก็บอยู่ในคอนเทนเนอร์เดียวกันทั้งหมด



รูปที่ 5.2 เอเจนต์ของระบบเฝ้าสังเกตข้อมูลบนหน้าเว็บไซต์บนเจดแพลตฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### บทสรุป

#### 6.1 สรุปโครงการ

โครงการนี้ได้ดำเนินการศึกษาข้อมูล ทฤษฎี และวิธีการนำเทคโนโลยีแบบมัลติเอเจนต์ และเอเจนต์เคลื่อนที่มาประยุกต์ใช้ในการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูล ซึ่งความรู้ที่ได้นี้สามารถนำมาเสนอแนวคิดในการออกแบบ และพัฒนาระบบการเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนหน้าเว็บไซด์ เพื่อศึกษาความเป็นไปได้ที่จะนำเทคโนโลยีการทำงานแบบมัลติเอเจนต์ให้เกิดประโยชน์ในแง่ของการทำงานร่วมกันของเอเจนต์ เอเจนต์แต่ละตัวที่อยู่ในระบบมีความเป็นอิสระ ขนาดเล็ก สามารถทำงานได้อย่างรวดเร็วและไม่ขึ้นต่อกัน ทั้งนี้ เพื่อทำให้เกิดประสิทธิภาพในการทำงานมากขึ้น และศึกษาความเป็นไปได้ที่จะนำเทคโนโลยีของเอเจนต์เคลื่อนที่มาช่วยในการเฝ้าสังเกตข้อมูลบนเว็บไซด์ เพื่อช่วยลดปริมาณการสื่อสารและขนาดของข้อมูลที่ใช้สื่อสารกันบนเครือข่าย นอกจากนี้ ระบบยังออกแบบให้มีเอเจนต์ที่คอยทำหน้าที่แจ้งเตือนกับผู้ใช้ผ่านแอปพลิเคชัน เมื่อข้อมูลบนหน้าเว็บไซด์มีการเปลี่ยนแปลง ช่วยทำให้ผู้ใช้รับทราบข่าวสารที่ตนเองสนใจได้อย่างรวดเร็ว

#### 6.2 ปัญหาของระบบงาน

การออกแบบในส่วนต่อประสานกับผู้ใช้งานยังไม่ค่อยมีประสิทธิภาพ เนื่องจากในส่วนของการตั้งค่าตัวคัดกรองที่จะทำการกรองข้อมูลในส่วนที่ผู้ใช้สนใจ ระบบยังไม่สามารถเลือกตัวคัดกรองได้มากกว่า 1 กลุ่มต่อเว็บไซด์

การแสดงผลการรายงานของระบบผ่านส่วนต่อประสานกับผู้ใช้งานไม่สามารถแสดงผลในรูปแบบที่มีรูปภาพได้

#### 6.3 ข้อจำกัดของระบบ

การพัฒนาระบบเฝ้าสังเกตการเปลี่ยนแปลงข้อมูลบนหน้าเว็บไซด์ ด้วยเทคโนโลยีการทำงานแบบมัลติเอเจนต์ และเอเจนต์เคลื่อนที่ ได้นำเสนอแนวคิดมาประยุกต์ใช้ในการเฝ้าสังเกตข้อมูลบนหน้าเว็บไซด์ ซึ่งยังคงมีข้อจำกัดดังนี้

1. เอเจนต์เซิร์ฟเวอร์ที่ต้องมีการนำเจดแพลตฟอร์มและเอเจนต์เซิร์ฟเวอร์ไปทำการติดตั้งยังเซิร์ฟเวอร์ เพื่อให้เอเจนต์เซิร์ฟเวอร์ทำการรวบรวมข้อมูลไว้ให้เอเจนต์เคลื่อนที่มาทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้าถึง ดังนั้น จึงต้องมีการจำลองเอเจนต์เซิร์ฟเวอร์ขึ้นมาแทนการทำงานบนเซิร์ฟเวอร์จริง

2. การเปรียบเทียบเว็บเพจควรแสดงหน้าเว็บไซต์เก่าและเว็บไซต์ใหม่ขึ้นมาเปรียบเทียบกัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

จามิกร หิรัญรัตน์. 2550. “โครงสร้างการทำงานโดยอาศัยเทคโนโลยีเอเจนต์แบบเคลื่อนที่สำหรับตลาดอิเล็กทรอนิกส์: กรณีศึกษา ตลาดกลางยางพาราของประเทศไทย”. [Online]. เข้าถึงได้จาก: [http://202.28.18.227/dcms/browse.php?option=show&browse\\_type=title&titleid=147563&query=จามิกร&s\\_mode=all&date\\_field=&date\\_start=&date\\_end=&limit\\_lang=&limited\\_lang\\_code=&order=&order\\_by=&order\\_type=&result\\_id=6&maxid=7](http://202.28.18.227/dcms/browse.php?option=show&browse_type=title&titleid=147563&query=จามิกร&s_mode=all&date_field=&date_start=&date_end=&limit_lang=&limited_lang_code=&order=&order_by=&order_type=&result_id=6&maxid=7)

เทอดพร สุโพธิ์, นิรันดร์กุล มนต์ขลัง และฉัตรตระกูล สมบัติธีระ. 2552. “ระบบมัลติเอเจนต์ภายในระบบโลจิสติกส์”. [Online]. เข้าถึงได้จาก: [http://italpha.msu.ac.th/csmis/seminar2009/data/BI/BI\\_04/ระบบมัลติเอเจนต์ภายในระบบโลจิสติกส์.doc](http://italpha.msu.ac.th/csmis/seminar2009/data/BI/BI_04/ระบบมัลติเอเจนต์ภายในระบบโลจิสติกส์.doc)

วิริยะ แก้วมรินทร์. 2550. **เทคโนโลยี Software Agent**. [Online]. เข้าถึงได้จาก: <http://202.28.94.51/users/Crawler-1/Slide-WorkShop-Crawler.pdf>

Bellifemine, Fabio, Caire, Giovanni, and Greenwood, Dominic. 2007. **Developing Multi-Agent Systems with JADE**. West Sussex : John Wiley & Sons.

Bellifemine, Fabio, Caire, Giovanni, Trucco, Tiziana, and Greenwood, Dominic. 2007a. **JADE Programmer’s Guide** [Online]. Available: <http://jade.tilab.com/doc/programmersguide.pdf>

Bellifemine, Fabio, Caire, Giovanni, Trucco, Tiziana, and Greenwood, Dominic. 2007b. **JADE Administrator’s Guide** [Online]. Available: <http://jade.tilab.com/doc/administratorsguide.pdf>

Darwin, Ian F. 2004. **Text File Difference Utility** [Online]. Available: <http://www.java2s.com/Code/Java/File-Input-Output/Differtextfiledifferenceutility.htm>

Hadzic, Maja, Dillon, Darshan S, and Dillon, Tharam S. 2009. “Use and Modeling of Multi-agent Systems in Medicine.” 303 – 307. **20th International Workshop on Database and Expert Systems Application**. Linz : Springer.

**Java Agent Development Framework**. 2010. [Online]. Available: [http://en.wikipedia.org/wiki/Java\\_Agent\\_Development\\_Framework](http://en.wikipedia.org/wiki/Java_Agent_Development_Framework)

Kelash, H. M., Amoon, M, Ali, G. M., and Faheem, H. M., 2005. “A Social Agent Interface for Resource Management in Distributed Systems.” 390-395. **Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Technologies and Internet Commerce. Vienna.**

Reynolds, Mark C., 1996. **“Java Programming From the Grounds Up”** [Online]. Available:

[http://www.webdeveloper.com/java/java\\_programming\\_grounds\\_up.html](http://www.webdeveloper.com/java/java_programming_grounds_up.html).



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติผู้เขียน

ชื่อผู้เขียน

นางมะลิวัลย์ เลาหกานต์นิคม

สถานที่เกิด

จังหวัดสิงห์บุรี

ประวัติการศึกษา

ระดับประถมศึกษา

โรงเรียนวัดโบสถ์ จังหวัดสิงห์บุรี

ระดับมัธยมศึกษาตอนต้น

โรงเรียนอินทร์บุรี จังหวัดสิงห์บุรี

ระดับมัธยมศึกษาตอนปลาย

โรงเรียนสิงห์บุรี จังหวัดสิงห์บุรี

ระดับอุดมศึกษา

วิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์)

คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเกษตรศาสตร์

ประวัติการทำงาน

พ.ศ. 2542 – มี.ย 2547

บริษัท สมาร์ทออฟฟิศ จำกัด

ก.ค. 2547 – ปัจจุบัน

บริษัท ที แมชชีนเนอร์รี่ (ประเทศไทย) จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้