

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รื้อชำ ระบบปฏิบัติการบนเว็บ

ROSA: REAL OPERATING SYSTEM APPLICATION



T117194



21พ.
ค1215
2553

เลขหมู่.....
เลขทะเบียน **117194**
ใน เดือน ปี **19 ก.ค. 2554**

b.....12341220.....
i.....

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROSA: REAL OPERATING SYSTEM APPLICATION

MISS KANOKPORN WANJITWONG

MR. CHOKCHAI PUTTAN



**A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIRMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
IN COMPUTER SCIENCE
FACULTY OF SCIENCE**

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ โรซ่า ระบบปฏิบัติการบนเว็บ
 ROSA: REAL OPERATING SYSTEM APPLICATION

ชื่อนักศึกษา นางสาวกนกพร วัลย์จิตรวงศ์ 50050097
 นายโชคชัย พุฒตาล 50050119

ปริญญา วิทยาศาสตรบัณฑิต
 สาขาวิชา วิทยาการคอมพิวเตอร์
 อาจารย์ที่ปรึกษา ผศ.ดร.ศรัณย์ อินทโกสุม

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2553

คณะกรรมการสอบ	ลายมือชื่อ
อ.วีระชัย ตันยะสิทธิ์ ประธานกรรมการ	
รศ.ดร.วีระ บุญจริง กรรมการ	
ผศ.ดร.ศรัณย์ อินทโกสุม กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ	โรซ่า ระบบปฏิบัติการบนเว็บ		
ชื่อนักศึกษา	นางสาวกนกพร	วัลย์จิตรวงศ์	50050097
	นายโชคชัย	พุ่มताल	50050119
ปริญญา	วิทยาศาสตรบัณฑิต		
สาขาวิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2553		
อาจารย์ที่ปรึกษา	ผศ.ดร.ศรัณย์ อินทโกสุม		

บทคัดย่อ

ระบบปฏิบัติการบนเว็บ หากพิจารณาแล้วจะหมายถึง ระบบปฏิบัติการที่ทำงานบนเว็บเบราว์เซอร์ และสามารถเรียกใช้ฟังก์ชันการทำงานขั้นพื้นฐานที่มีในระบบปฏิบัติการบนเดสก์ท็อปได้ เช่น ระบบปฏิบัติการวินโดวส์ และ ลินุกซ์ โดยแนวคิดที่เป็นจุดเด่นของระบบปฏิบัติการบนเว็บคือ ผู้ใช้ไม่จำเป็นต้องใช้อุปกรณ์คอมพิวเตอร์ที่มีความสามารถสูง และผู้ใช้สามารถเข้าถึงโปรแกรมและข้อมูลของตนเองจากที่ใดก็ได้ที่มีอินเทอร์เน็ต ระบบปฏิบัติการบนเว็บนั้นสามารถนำไปใช้กับองค์กรที่จำเป็นต้องมีการควบคุมความปลอดภัยของข้อมูล และต้องการลดค่าใช้จ่ายของอุปกรณ์คอมพิวเตอร์ของผู้ใช้ในองค์กร โครงการพิเศษนี้คือการพัฒนาโรซ่า ระบบปฏิบัติการบนเว็บที่มุ่งเน้นไปที่การนำไปใช้งานภายในองค์กร โดยโรซ่าได้ถูกออกแบบและพัฒนาให้สนับสนุนด้านการรักษาความปลอดภัยของข้อมูล และง่ายในการพัฒนาต่อยอด โรซ่านั้นประกอบด้วย ระบบจัดการข้อมูล ระบบจัดการแอปพลิเคชัน และระบบบริหารจัดการระบบ ซึ่งผู้ดูแลระบบสามารถปรับแต่งระบบให้เหมาะสมกับนโยบาย และการใช้งานภายในองค์กรได้ นอกจากนี้ระบบยังมีการจัดเตรียม API ให้กับนักพัฒนา เพื่อให้ นักพัฒนาสามารถพัฒนาแอปพลิเคชัน ได้ง่าย

คำสำคัญ : เว็บแอปพลิเคชัน เว็บ โอเอส ระบบปฏิบัติการบนเว็บ ระบบปฏิบัติการ

Title	ROSA: REAL OPERATING SYSTEM APPLICATION		
Students	Ms.Kanokporn	Wanjitwong	50050097
	Mr.Chokchai	Puttan	50050119
Degree	Bachelor of Science		
Major Program	Computer Science		
Academic Year	2010		
Advisor	Asst.Prof.Dr.Sarun Intakosum		

ABSTRACT

A web operating system can be considered an operating system that runs on web browsers and can perform basic functions of desktop operating systems, such as Windows and Linux operating systems. The major idea of the web operating system are that it does not require high performance clients and users can access their programs and data from everywhere that has the internet access. Web operating system can also be used in an organization that needs to control the security of data and reduce the cost of client computers in the organization. This special project develops ROSA, a web operating system that aims to be used in an organization. It is designed and developed to support data security and can be easily extended. ROSA comprises of Data Storage Management System, Application Management System, and Administration System. Administrator can configure system settings to suit an organization policy and usage. In addition, the system also provides developers with API to simplify the development of applications.

Keywords : Web Application, Web OS, Web Operating System, Operating System

กิตติกรรมประกาศ

การทำโครงการพิเศษหัวข้อเรื่อง โรซ่า ระบบปฏิบัติการบนเว็บไซต์ฉบับนี้สำเร็จลงได้ด้วยดีจากการช่วยเหลือและสนับสนุนของบุคคลหลายท่าน คณะผู้จัดทำขอขอบพระคุณบุคคลดังต่อไปนี้

1. บิดา มารดา สมาชิกในครอบครัว ผู้ให้การอบรมสั่งสอน คอยสนับสนุนด้านการศึกษา และให้กำลังใจเสมอมา
2. ผศ.ดร.ศรัณย์ อินทโกสุม อาจารย์ที่ปรึกษาโครงการพิเศษ ที่กรุณาให้คำปรึกษาและแนะนำแนวทางการปฏิบัติงานเสมอมา
3. อาจารย์วีระชัย ต้นยะสิทธิ์ และรศ.ดร.วีระ บุญจริง ประธานและกรรมการ โครงการพิเศษ ที่กรุณาเป็นกรรมการคุมสอบ
4. อาจารย์สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความรู้ตลอดระยะเวลา 4 ปี
5. เพื่อนๆ ในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่คอยให้กำลังใจฝ่าฟันอุปสรรคในการทำโครงการพิเศษ

นอกจากนี้ยังมีบุคคลท่านอื่นๆ ที่ไม่ได้กล่าวไว้ ณ ที่นี้ จึงใคร่ขอขอบพระคุณทุกท่านที่มีส่วนร่วมในการให้ความช่วยเหลือ คำปรึกษา คำแนะนำ ตลอดจนกำลังใจในการจัดทำโครงการพิเศษฉบับนี้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงการพิเศษ	1
1.2 วัตถุประสงค์ของโครงการพิเศษ	2
1.3 ขอบเขตและข้อจำกัดของโครงการพิเศษ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.1.1 ระบบปฏิบัติการบนเว็บ (Web Operating System)	4
2.1.2 เทคโนโลยีเอแจ็กซ์ (Ajax)	5
2.2 งานวิจัยที่เกี่ยวข้อง	7
2.2.1 อายโอเอส (eyeOS)	7
2.2.1.1 คุณสมบัติของอายโอเอส	8
2.2.1.2 สถาปัตยกรรมของอายโอเอส	12

บทที่ 3 การวิเคราะห์และออกแบบระบบ

3.1 การวิเคราะห์ระบบ	15
3.1.1 วิเคราะห์การใช้งาน	15
3.1.1.1 ผู้ใช้ (User)	16
3.1.1.2 ผู้ดูแลระบบ (Administrator)	16
3.1.1.3 นักพัฒนา (Developer)	16
3.1.2 วิเคราะห์โครงสร้างการทำงาน	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.1.2.1 ไคลเอนต์ (Client)	17
3.1.2.2 เซิร์ฟเวอร์ (Server)	17
3.1.3 วิเคราะห์ระบบไฟล์	18
3.1.3.1 ความปลอดภัย	18
3.1.3.2 ความรวดเร็วในการรวบรวมข้อมูล	18
3.1.3.3 ความสะดวกในการเข้าถึง	18
3.2 การออกแบบระบบ	19
3.2.1 สถาปัตยกรรม	19
3.1.3.1 โรซ่าไคลเอนต์ (Rosa Client)	20
3.1.3.2 โรซ่าเซิร์ฟเวอร์ (Rosa Server)	21
3.2.2 การออกแบบระบบไฟล์	22
3.2.3 การออกแบบส่วนติดต่อกับผู้ใช้	24
3.2.3.1 เดสก์ท็อป (Desktop)	24
3.2.3.2 ป๊อปอัพเมนู (Popup menu หรือ Right-click menu)	25
3.2.3.3 หน้าต่าง (Window)	25
3.2.3.4 ทาสก์บาร์ (Taskbar)	26
3.2.3.5 สตาร์ทเมนู (Start menu)	26
3.2.4 การออกแบบขั้นตอนในการพัฒนาและการนำแอปพลิเคชันไปใช้งาน	27
บทที่ 4 ผลการดำเนินงาน	
4.1 ภาพรวมของระบบ	29
4.2 ความสามารถของระบบ	29
4.2.1 ความสามารถของระบบในส่วนของผู้ใช้ระบบ	29
4.2.1.1 การสร้างบัญชีผู้ใช้งาน	29
4.2.1.2 การตรวจสอบยืนยันตัวตนผู้ใช้	30
4.2.1.3 การจัดการไฟล์บนเดสก์ท็อปและเอ็กซ์พลอเรอร์	31
4.2.1.4 แอปพลิเคชันพื้นฐาน	33
4.2.1.5 การล็อกเข้าที่ออกจากระบบ	40
4.2.2 ความสามารถของระบบในส่วนของผู้ดูแลระบบ	40

สารบัญ (ต่อ)

	หน้า
4.2.2.1 การตรวจสอบสิทธิของผู้ดูแลระบบ	41
4.2.2.2 การจัดการข้อมูลสำหรับผู้ดูแลระบบ	42
4.2.2.3 การล็อกเอาต์ออกจากระบบ	46
4.2.3 การพัฒนาแอปพลิเคชันสำหรับ โรซ่าไอเอส	46
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	
5.1 สรุปผลการดำเนินงาน	52
5.2 ข้อเสนอแนะ	52
เอกสารอ้างอิง	54
ภาคผนวก	แผ่น CD



สารบัญรูป

รูปที่	หน้า
2.1 เปรียบเทียบการติดต่อสื่อสารระหว่างเว็บแอปพลิเคชันแบบดั้งเดิมกับแบบที่ใช้เอเจ็กซ์	6
2.2 ส่วนติดต่อกับผู้ใช้ของอายุโอเอส	7
2.3 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันสำหรับจัดการไฟล์ในอายุโอเอส	9
2.4 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันปฏิทินส่วนตัวในอายุโอเอส	10
2.5 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันประมวลผลคำในอายุโอเอส	10
2.6 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันสำหรับอ่าน RSS ในโรซ่าโอเอส	11
2.7 สถาปัตยกรรมของอายุโอเอส	12
2.8 ตัวอย่างส่วนติดต่อกับผู้ใช้ของอายุโอเอสเมื่อใช้งานผ่านบราวเซอร์	14
2.9 ตัวอย่างส่วนติดต่อกับผู้ใช้ของอายุโอเอสเมื่อใช้งานผ่านโทรศัพท์มือถือ	14
3.1 ยูสเคสแสดงการใช้งานโรซ่าโอเอส	15
3.2 สถาปัตยกรรมของโรซ่าโอเอส	19
3.3 โครงสร้างระบบโรซ่าโอเอสทางฝั่งไคลเอนต์	20
3.4 โครงสร้างระบบโรซ่าโอเอสทางฝั่งเซิร์ฟเวอร์	21
3.5 ระบบไฟล์ของโรซ่าโอเอส	22
3.6 ตัวอย่างการทำงานของระบบไฟล์ เมื่อผู้ใช้อัปโหลดไฟล์เข้าสู่ระบบ	24
3.7 การเลือกไฟล์มากกว่าหนึ่งไฟล์บนเดสก์ท็อป	25
3.8 ป๊อปอัพเมนู	25
3.9 หน้าต่างภายในโรซ่าโอเอส	26
3.10 ทาสก์บาร์	26
3.11 สตาร์ทเมนู	27
3.12 การพัฒนาแอปพลิเคชันและการนำแอปพลิเคชัน ไปใช้งาน	28
4.1 การสร้างบัญชีผู้ใช้งาน	30
4.2 หน้าจอการล็อกอินเข้าสู่ระบบของผู้ใช้ทั่วไป	30
4.3 หน้าจอเดสก์ท็อป	31
4.4 หน้าจอเอ็กซ์พลอเรอร์	32
4.5 เมนูคำสั่งที่ปรากฏขึ้นเมื่อคลิกเมาส์ขวาที่ไอคอนของไฟล์หรือโฟลเดอร์	33
4.6 แอปพลิเคชันพื้นฐานที่รับจัดเตรียมไว้ให้	34
4.7 Firebug Lite	35
4.8 Rosa TaskManager	36

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.9 การรันแอปพลิเคชันผ่านทาง Rosa TaskManager	36
4.10 Rosa MusicPlayer	37
4.11 Rosa Notepad	38
4.12 Rosa MoviePlayer	39
4.13 การลือกเอาที่ออกจากกระบบ	40
4.14 หน้าจอการลือกอินเข้าสู่ระบบของผู้ดูแลระบบ	41
4.15 หน้าจอการใช้งานของส่วนของผู้ดูแลระบบ	42
4.16 ส่วนของข้อมูลที่เกี่ยวข้องกับผู้ใช้ในหน้าจอการใช้งานของผู้ดูแลระบบ	43
4.17 การแก้ไขข้อมูลที่เกี่ยวข้องกับผู้ใช้ในหน้าจอการใช้งานของผู้ดูแลระบบ	44
4.18 ส่วนของข้อมูลที่เกี่ยวข้องกับแอปพลิเคชันในหน้าจอการใช้งานของผู้ดูแลระบบ	45
4.19 การลือกเอาที่ออกจากส่วนของผู้ดูแลระบบ	46
4.20 ต้นแบบส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน SayHello	47
4.21 โครงสร้างไฟล์ของแอปพลิเคชัน SayHello	47
4.22 ซอร์สโค้ดของไฟล์ say_hello.json	48
4.23 ผลที่ได้จากการเขียนไฟล์สำหรับอธิบายแอปพลิเคชันของแอปพลิเคชัน SayHello	49
4.24 ซอร์สโค้ดของแอปพลิเคชันออบเจกต์ที่ได้ลงทะเบียนกับส่วนจัดการแอปพลิเคชันแล้ว	49
4.25 ซอร์สโค้ดเพิ่มเติมฟังก์ชันการทำงานที่อยู่ในแอปพลิเคชันออบเจกต์	50
4.26 ส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน SayHello	51

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการพิเศษ

เว็บแอปพลิเคชันนั้นเป็นแอปพลิเคชันในรูปแบบหนึ่งที่มีความสามารถในการทำงานใกล้เคียงกับโปรแกรมคอมพิวเตอร์ที่ติดตั้งบนเดสก์ท็อปแต่มีการใช้เว็บเบราว์เซอร์ในการแสดงผล โดยผู้ใช้สามารถเข้าใช้งานเว็บแอปพลิเคชันได้จากทุกแพลตฟอร์ม (Platform) ที่มีเว็บเบราว์เซอร์และอินเทอร์เน็ต เนื่องจากปัจจุบันการใช้งานเทคโนโลยีอินเทอร์เน็ตความเร็วสูงมีจำนวนเพิ่มขึ้นอย่างต่อเนื่อง อีกทั้งเว็บเบราว์เซอร์ก็มีความสามารถมากขึ้น จึงส่งผลให้เว็บแอปพลิเคชันได้รับความนิยมมากขึ้น และแนวคิดในการพัฒนาระบบปฏิบัติการที่ทำงานบนเว็บเบราว์เซอร์นั้นได้รับความนิยมมากขึ้นด้วย

ระบบปฏิบัติการที่ทำงานบนเว็บเบราว์เซอร์ หรือ “เว็บโอเอส” (Web Operating System : WebOS) คือแอปพลิเคชันที่มีความสามารถในการทำงานพื้นฐานคล้ายกับระบบปฏิบัติการบนเดสก์ท็อปที่ผู้ใช้คุ้นเคย ตัวอย่างการพัฒนาเว็บโอเอสในอดีตที่ผ่านมาเช่น อายโอเอส (eyeOS), ยูโอเอส (YouOS), โกส (G.ho.st), ไกลด์โอเอส (Glide OS) และ ไอคลาวด์ (icloud) แต่ปัจจุบันเหลือผู้ให้บริการเพียงแค่บางรายเท่านั้น โดยสาเหตุที่เว็บโอเอสบางรายไม่ได้รับความนิยมก็คือ ไม่มีการจัดการการใช้งานทรัพยากรที่ดี ขาดการจัดการควบคุมการใช้งานของผู้ใช้ที่ดี และไม่สามารถทำงานข้ามแพลตฟอร์มได้อย่างแท้จริง (มีการใช้ Adobe Flash) ทั้งประกอบกับการที่ในยุคสมัยนี้ความเร็วและความทั่วถึงของอินเทอร์เน็ตยังไม่เพียงพอกับความต้องการของเว็บโอเอสดังเช่นปัจจุบันนี้

จากเหตุผลดังกล่าวจึงได้จัดทำเว็บโอเอสเพื่อหลีกเลี่ยงปัญหาข้างต้น โดยมีจุดประสงค์หลักในการนำไปใช้งานภายในองค์กร เพื่อให้เกิดความปลอดภัยทางด้านข้อมูลภายในองค์กร และสนับสนุนการนำไปพัฒนาต่อยอด โดยเว็บโอเอสนี้ใช้ชื่อว่า "โรซ่า" (ROSA : Real Operating System Applications) หรือ “โรซ่าโอเอส” ซึ่งโรซ่าโอเอสมีเป้าหมายคือให้ผู้ใช้สามารถนำไปใช้งานได้ในเบื้องต้น และมีการจัดเตรียมแอปพลิเคชันพื้นฐานของระบบปฏิบัติการไว้ให้ผู้ใช้ นอกจากนี้เพื่อให้การนำระบบไปพัฒนาต่อยอดสามารถทำได้สะดวกจึงได้มีการจัดเตรียม API (Application Programming Interface) ไว้ให้กับนักพัฒนาอีกด้วย

1.2 วัตถุประสงค์ของโครงการพิเศษ

- 1) เพื่อพัฒนาเว็บ โอเอสที่มีความสามารถของระบบปฏิบัติการขั้นพื้นฐาน ได้แก่ ระบบจัดการข้อมูลของผู้ใช้ ระบบจัดการแอปพลิเคชันสำหรับผู้ใช้ และระบบสำหรับผู้ดูแลระบบ
- 2) เพื่อพัฒนาเว็บ โอเอสสำหรับการใช้งานภายในองค์กร เพื่อให้เกิดความปลอดภัยทางด้านข้อมูลภายในองค์กร
- 3) เพื่อพัฒนาเว็บ โอเอสที่สนับสนุนการนำไปพัฒนาต่อยอด

1.3 ขอบเขตและข้อจำกัดของโครงการพิเศษ

- 1) สามารถนำไปใช้งานได้ในพื้นที่พื้นฐานเท่านั้น โดยมีระบบจัดการข้อมูลของผู้ใช้ ระบบจัดการแอปพลิเคชันสำหรับผู้ใช้ และระบบสำหรับผู้ดูแลระบบ ในการใช้งานผู้ใช้จะได้รับพื้นที่การใช้งานส่วนตัวและสามารถใช้งานแอปพลิเคชันพื้นฐานที่ระบบจัดเตรียมไว้ให้ได้ ผู้ดูแลระบบสามารถจัดการข้อมูลเกี่ยวกับผู้ใช้และแอปพลิเคชันได้ และนักพัฒนาสามารถพัฒนาแอปพลิเคชันเพิ่มเติมขึ้นมาใช้งานบนโรซ่าได้
 - 2) ผู้ใช้ไม่สามารถติดตั้งซอฟต์แวร์ที่พัฒนาขึ้นมาเพื่อประมวลผลบนระบบปฏิบัติการบนเดสก์ท็อป เช่น วินโดวส์ ลงบนโรซ่าโอเอสได้
 - 3) ไม่มีความสามารถในการจัดการ ไร้เซิร์ฟเวอร์ และไม่สามารถติดต่อกับเซิร์ฟเวอร์ได้
 - 4) ต้องการเน็ตเวิร์กที่มีความเร็วอย่างต่ำ 128 Kbps ระบบจึงจะสามารถทำงานได้อย่างราบรื่น
- สนับสนุนการทำงานบนเว็บเบราว์เซอร์เพียงแค่บางเว็บเบราว์เซอร์เท่านั้น คือ Firefox เวอร์ชัน 3.5 ขึ้นไป และ Google Chrome

1.4 ขั้นตอนการดำเนินงาน

- 1) ศึกษาหาข้อมูลสำหรับการพัฒนาเว็บ โอเอส
- 2) กำหนดขอบเขต จุดประสงค์ และความต้องการของระบบ
- 3) ออกแบบโครงสร้างการทำงาน และส่วนติดต่อกับผู้ใช้ของโรซ่า
- 4) พัฒนาส่วนการทำงานหลัก (Kernel) และไลบรารี (Libraries) หรือโรซ่าเซิร์ฟเวอร์
- 5) พัฒนาส่วนควบคุมการทำงานของแอปพลิเคชันหรือโรซ่าไคลเอนต์
- 6) พัฒนาส่วนติดต่อกับผู้ใช้ (User Interface)
- 7) พัฒนา API ของแอปพลิเคชัน
- 8) พัฒนาส่วน Administration ของโรซ่า
- 9) พัฒนาแอปพลิเคชันพื้นฐานของโรซ่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 10) ทดสอบระบบ และปรับปรุงแก้ไขข้อผิดพลาด
- 11) จัดทำเอกสารปัญหาพิเศษ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ผู้ใช้สามารถนำโรซ่าโอเอสไปใช้งานได้ในขั้นพื้นฐาน
- 2) ผู้ใช้สามารถนำโรซ่าโอเอสไปใช้งานในองค์กร เพื่อให้เกิดความปลอดภัยทางด้านข้อมูลภายในองค์กร
- 3) ผู้พัฒนาสามารถนำโรซ่าโอเอสไปพัฒนาต่อยอดได้ ทั้งผ่านทางเอพีไอ และซอร์สโค้ด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ระบบปฏิบัติการบนเว็บ

ระบบปฏิบัติการบนเว็บหรือเว็บ โอเอส (Web OS) เป็นแอปพลิเคชันที่ให้บริการผ่านทางเว็บเบราว์เซอร์ชนิดหนึ่งโดยใช้เว็บเบราว์เซอร์เป็นส่วนติดต่อผู้ใช้ และมีความสามารถพื้นฐานคล้ายคลึงกับระบบปฏิบัติการบนเดสก์ท็อป เช่น มีระบบจัดการไฟล์ และมีแอปพลิเคชันพื้นฐานที่ให้บริการ เช่น โปรแกรมประมวลผลคำ โปรแกรมเครื่องคิดเลข โปรแกรมดูรูปภาพ และโปรแกรมเล่นไฟล์เสียงหรือวิดีโอ เป็นต้น

จุดเด่นของเว็บ โอเอสคือการให้บริการผ่านทางอินเทอร์เน็ต ซึ่งส่งผลให้ผู้ใช้สามารถลงชื่อเข้าใช้บริการได้จากทั่วทุกมุมโลกผ่านทางอุปกรณ์คอมพิวเตอร์ใดๆที่มีเว็บเบราว์เซอร์และสามารถเชื่อมต่อกับอินเทอร์เน็ตได้ อีกทั้งเว็บ โอเอสยังมีความเป็นอิสระจากแพลตฟอร์ม กล่าวคือผู้ใช้สามารถใช้บริการเว็บ โอเอสได้จากทุกแพลตฟอร์มอีกด้วย

ในอดีตเว็บ โอเอสไม่ได้รับความนิยม เนื่องจากมีข้อจำกัดทางด้านความเร็วอินเทอร์เน็ตของผู้ใช้ ทำให้การทำงานและการให้บริการของเว็บ โอเอสมีความรวดเร็วไม่เพียงพอ ส่งผลให้ผู้ใช้ไม่ได้รับความพึงพอใจจากการใช้บริการมากเท่าที่ควร แต่ปัจจุบันเทคโนโลยีอินเทอร์เน็ตมีการพัฒนาขึ้นมาก ทำให้มาตรฐานความเร็วอินเทอร์เน็ตของผู้ใช้มีระดับที่สูงขึ้น เว็บ โอเอสจึงกลับมาได้รับความสนใจกันอย่างกว้างขวางมากขึ้น

นอกเหนือจากข้อดีที่ผู้ใช้สามารถลงชื่อเข้าใช้งานเว็บ โอเอสได้จากทั่วทุกมุมโลกแล้ว ผู้ใช้ยังได้รับประโยชน์ในข้ออื่น ๆ อีก เช่น สามารถทำงานร่วมกับผู้ใช้คนอื่นๆในรูปแบบออนไลน์ได้ และยังสามารถฟื้นคืนข้อมูลที่สูญหาย ทำให้สามารถนำไฟล์ต่างๆมาเก็บสำรอง (Backup) ไว้ในส่วนนี้ได้

2.1.2 เทคโนโลยีเอแจ็กซ์ (Ajax)

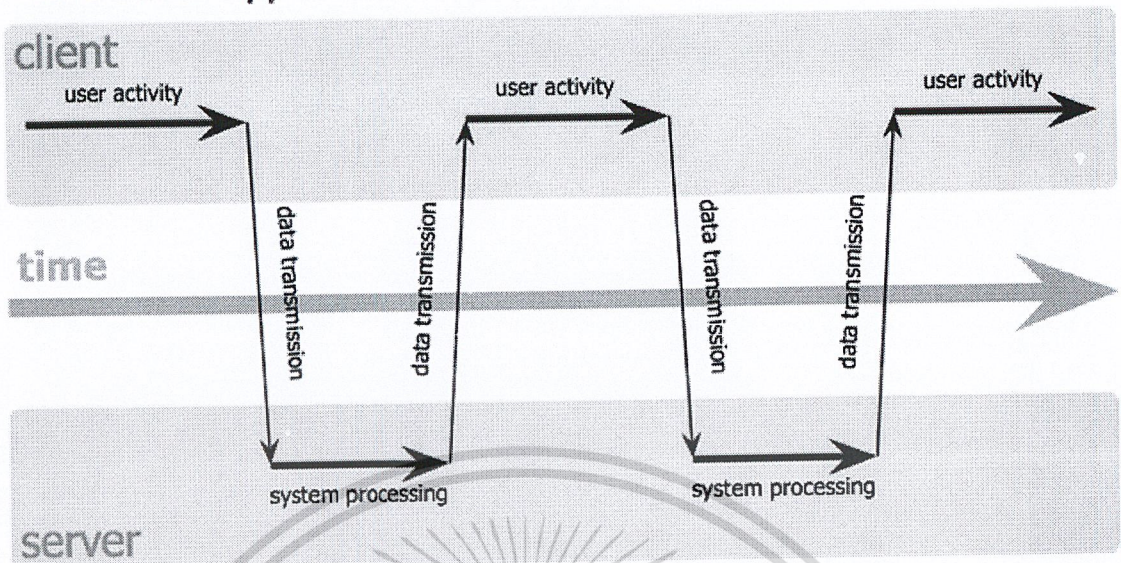
เอแจ็กซ์ คือเทคนิคที่ใช้ในการพัฒนาเว็บแอปพลิเคชันเพื่อให้เว็บแอปพลิเคชันสามารถโต้ตอบกับผู้ใช้ได้ดีขึ้น โดยจะรับส่งข้อมูลในเบื้องหลัง ทำให้ไม่จำเป็นต้องร้องขอข้อมูลใหม่ทุกครั้งที่มีการเปลี่ยนแปลง ซึ่งช่วยให้ผู้ใช้สามารถตอบสนองการทำงานได้อย่างรวดเร็ว และรู้สึกสะดวกสบายมากขึ้น จึงนับว่าเป็นเทคโนโลยีสำคัญที่นำมาใช้ในการพัฒนาเว็บโอเอส

เอแจ็กซ์นั้นไม่ใช่เทคโนโลยีใหม่ แต่เป็นเทคนิคที่ได้ใช้เทคโนโลยีที่มีอยู่แล้วรวมกันดังนี้

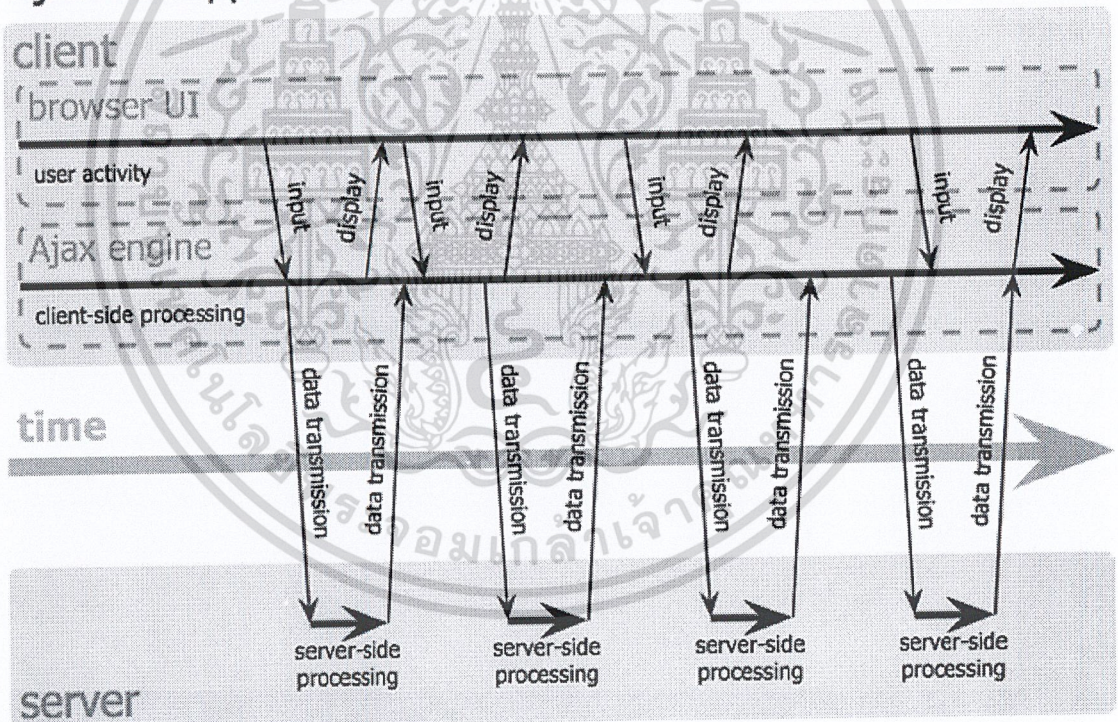
- HTML และ CSS ใช้ในการแสดงผลลัพธ์และรูปแบบข้อมูล
- DOM (Document Object Model) ใช้ในการแสดงข้อมูลที่มีการเปลี่ยนแปลงหรือโต้ตอบกับผู้ใช้
- XML HTML หรือ JSON ใช้เป็นรูปแบบข้อมูลในการแลกเปลี่ยน
- XMLHttpRequest ใช้ในการแลกเปลี่ยนข้อมูลแบบไม่ต่อเนื่อง (Asynchronous) กับเว็บเซิร์ฟเวอร์
- จาวาสคริปต์ สำหรับนำเทคโนโลยีที่กล่าวมาทั้งหมดมาทำงานร่วมกัน

จากรูปที่ 2.1 แสดงการเปรียบเทียบระหว่างการติดต่อสื่อสารแบบต่อเนื่อง (Synchronous) หรือไม่ใช่ Ajax และการติดต่อสื่อสารแบบไม่ต่อเนื่อง (Asynchronous) หรือใช้ Ajax จะเห็นว่าการติดต่อสื่อสารแบบต่อเนื่อง (รูปบน) นั้น ขณะที่ผู้ใช้ที่ไคลเอนต์มีกิจกรรมกับระบบ (user activity) และต้องการร้องขอไปยังเซิร์ฟเวอร์ จึงได้ร้องขอและส่งข้อมูลการร้องขอไปยังเซิร์ฟเวอร์ (data transmission) จะสังเกตเห็นว่ากิจกรรมของผู้ใช้หยุดลงในระหว่างที่รอเซิร์ฟเวอร์ประมวลผลการร้องขอ (system processing) ซึ่งจะเริ่มกิจกรรมได้อีกครั้งเมื่อเซิร์ฟเวอร์ประมวลผลเสร็จสิ้นและส่งข้อมูลกลับมา (data transmission) แต่หากใช้การติดต่อแบบไม่ต่อเนื่อง (Asynchronous) ผู้ใช้ที่ไคลเอนต์จะสามารถทำกิจกรรมอย่างได้ต่อเนื่อง เนื่องจากมีการสร้าง Ajax Engine เพื่อเป็นตัวแทนในการร้องขอข้อมูลจากเซิร์ฟเวอร์ กล่าวคือ เมื่อผู้ใช้ใช้งานส่วนติดต่อกับผู้ใช้ของบราวเซอร์ (Browser UI) และต้องการร้องขอข้อมูลจากเซิร์ฟเวอร์ ส่วนติดต่อกับผู้ใช้ของบราวเซอร์จะทำการส่งข้อมูลที่ต้องการร้องขอ (input) ไปยัง Ajax Engine จากนั้น Ajax Engine จะทำการนำข้อมูลดังกล่าวร้องขอไปยังเซิร์ฟเวอร์ต่อไป โดยที่ในขณะที่ Ajax Engine ทำการร้องขอนั้น ผู้ใช้สามารถดำเนินกิจกรรมได้อย่างต่อเนื่อง และเมื่อเซิร์ฟเวอร์ทำการประมวลผลเสร็จสิ้นก็จะส่งข้อมูลกลับมายัง Ajax Engine จากนั้น Ajax Engine จะนำข้อมูลที่ได้นำไปแสดงผล (display) ที่ส่วนติดต่อกับผู้ใช้ของบราวเซอร์ต่อไป

classic web application model (synchronous)



Ajax web application model (asynchronous)



Jesse James Garrett / adaptivepath.com

รูปที่ 2.1 เปรียบเทียบการติดต่อสื่อสารระหว่างเว็บแอปพลิเคชันแบบดั้งเดิมกับแบบที่ใช้เอแจ็กซ์

(<http://www.adaptivepath.com/ideas/essays/archives/000385.php>)

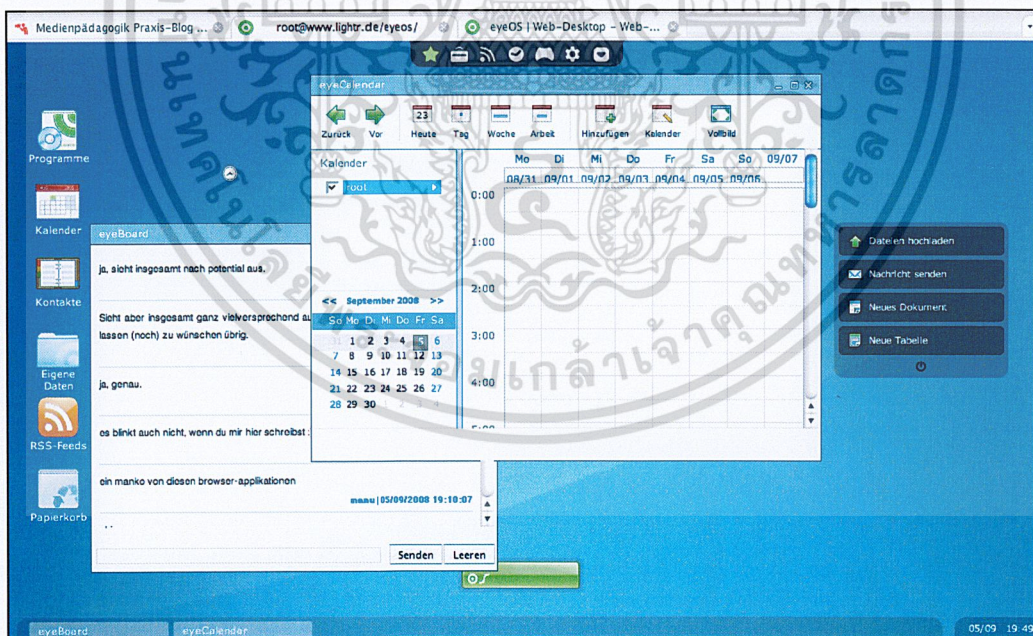
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 งานวิจัยที่เกี่ยวข้อง

2.1.1 อายโอเอส (eyeOS)

เป็นระบบปฏิบัติการโอเพ่นซอร์สบนเว็บที่ถูกพัฒนาขึ้นตามแนวคิดของการประมวลผลแบบคลาวด์ในปี พ.ศ.2548 โดยกลุ่มนักพัฒนาชาวสเปน อายโอเอสมีบทบาทสำคัญในการให้ความหมายของคำว่าระบบปฏิบัติการบนเว็บ และถือเป็นระบบปฏิบัติการบนเว็บที่ได้รับความนิยมมากที่สุดในปัจจุบัน

อายโอเอสนั้น เป็นเว็บแอปพลิเคชันที่สร้างขึ้นมาจากมีส่วนติดต่อผู้ใช้คล้ายคลึงระบบปฏิบัติการทั่วไป (Windows, OSX หรือ Ubuntu) โดยการนำเอาหลักการของระบบปฏิบัติการมาใช้ในออกแบบและพัฒนา ซึ่งประกอบไปด้วยหลายส่วนด้วยกัน ไม่ว่าจะเป็นเรื่องของการทำงานที่มีระบบควบคุมการเข้าถึงข้อมูล (Access control) มีความสามารถในการสร้างกลุ่มและจัดการกลุ่มผู้ใช้ สามารถแบ่งการใช้งานพื้นที่ของผู้ใช้ได้ มีระบบแคช (Cache) มีระบบจัดการไฟล์ มีระบบเริ่มต้นการทำงานแอปพลิเคชันโดยอัตโนมัติเมื่อเริ่มใช้งาน สามารถปรับแต่งส่วนติดต่อผู้ใช้ได้ และมีการเพิ่มเติมความสามารถในการทำงานร่วมกับ OpenOffice (ชุดแอปพลิเคชันสำหรับสำนักงาน) ซึ่งจะเห็นว่าอายโอเอสนั้นมีการเตรียมระบบไว้พร้อมสำหรับการทำงานได้ในระดับที่ใกล้เคียงกับระบบปฏิบัติการ โดยสามารถแสดงส่วนติดต่อกับผู้ใช้ของอายโอเอสได้ ดังรูปที่ 2.2



รูปที่ 2.2 ส่วนติดต่อกับผู้ใช้ของอายโอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) การจัดการไฟล์ (File Management)

การจัดการไฟล์ของอายุโอเอส นั้นสามารถควบคุมการใช้งานไฟล์ของผู้ใช้แต่ละคนได้อย่างสมบูรณ์ โดยผู้ใช้จะไม่สามารถเข้าถึงไฟล์ของผู้อื่นได้ แต่ผู้ใช้สามารถแบ่งปันไฟล์ให้กับผู้อื่นผ่านระบบการแบ่งปันได้ ซึ่งความสามารถด้านการใช้งานของส่วนจัดการไฟล์นั้นมีดังต่อไปนี้

- สามารถอัปโหลดไฟล์ได้ครั้งละหลายไฟล์
- มีระบบจัดการไฟล์ที่มีส่วนติดต่อผู้ใช้ในการสร้าง แก้ไข และลบไฟล์ ดังรูปที่ 2.3
- สามารถดาวน์โหลดไฟล์เดสก์ทอปทั้งไฟล์เดสก์ทอปในรูปแบบของชิปไฟล์ได้
- สามารถรวมไฟล์และแยกไฟล์ด้วยชิปได้
- สามารถแสดงรูปภาพในรูปแบบสไลด์โชว์ได้

2) การจัดการข้อมูลส่วนตัวของผู้ใช้ (Personal Information Management)

อายุโอเอส นั้นมีแอปพลิเคชันสำหรับจัดการข้อมูลเกี่ยวกับผู้ใช้ ดังต่อไปนี้

- ปฏิทินส่วนตัว ดังรูปที่ 2.4
- สามารถอิมพอร์ตและเอ็กซ์พอร์ตวีการ์ด (vCard) ได้

3) แอปพลิเคชันสำหรับสำนักงาน (Office)

อายุโอเอสมีแอปพลิเคชันสำหรับสำนักงาน โดยสามารถทำงานร่วมกับไฟล์ที่ถูกสร้างขึ้นบนแอปพลิเคชันสำหรับสำนักงานทั่วไป (เช่น Microsoft Office และ OpenOffice) ดังนี้

- แอปพลิเคชันสำหรับประมวลผลคำและตารางคำนวณ (Word processor and Spreadsheets) ดังรูปที่ 2.5
- แอปพลิเคชันสำหรับสร้างพรีเซนเทชัน (Presentations)

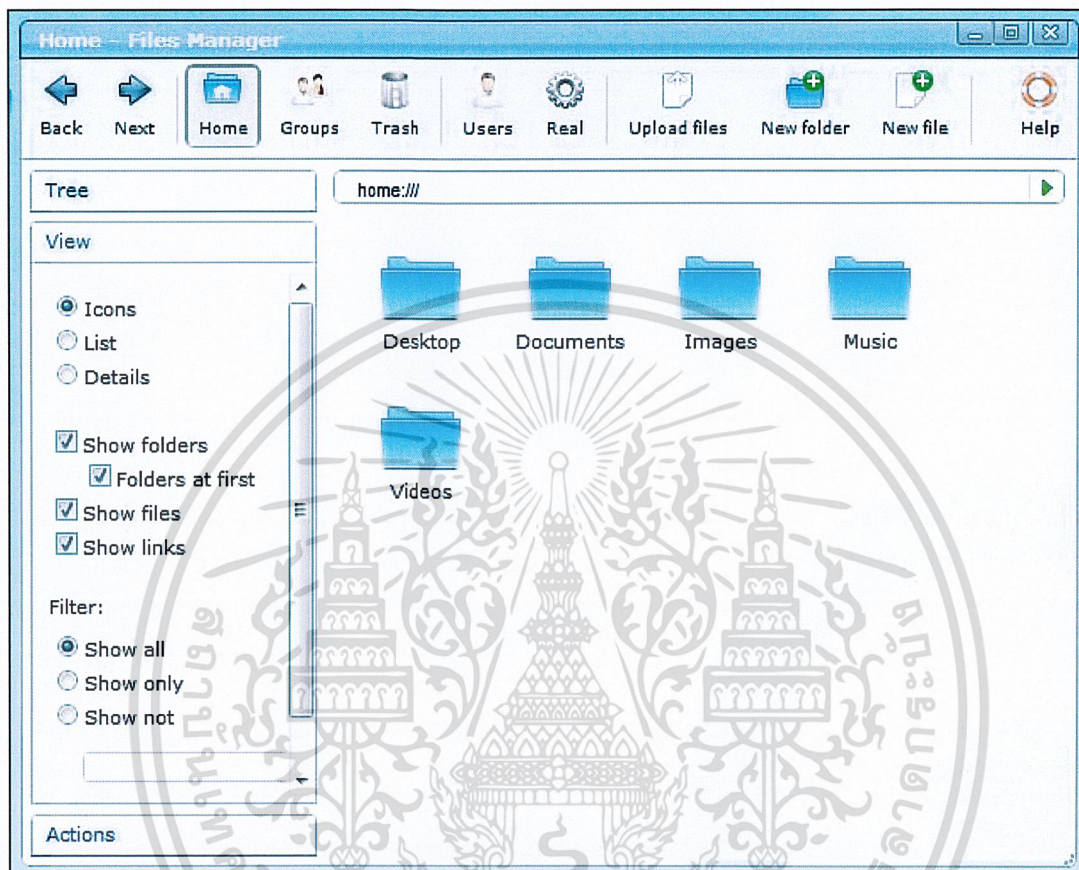
4) เน็ตเวิร์ก (Network)

อายุโอเอสมีแอปพลิเคชันสำหรับทำงานที่เกี่ยวข้องกับเน็ตเวิร์ก โดยจะมีทั้งแอปพลิเคชันที่เป็นการติดต่อภายในอายุโอเอสเอง และแอปพลิเคชันที่ติดต่อกับโลกภายนอก ซึ่งแอปพลิเคชันที่เกี่ยวข้องกับเน็ตเวิร์กของอายุโอเอสมีดังต่อไปนี้

- อีเมลไคลเอนต์และอินเทอร์เนตเมลเซจจิง (Email client and Internal Messaging)
- เอฟทีพีไคลเอนต์ (FTP client)
- แอปพลิเคชันในการอ่าน RSS (RSS news reader) ดังรูปที่ 2.6

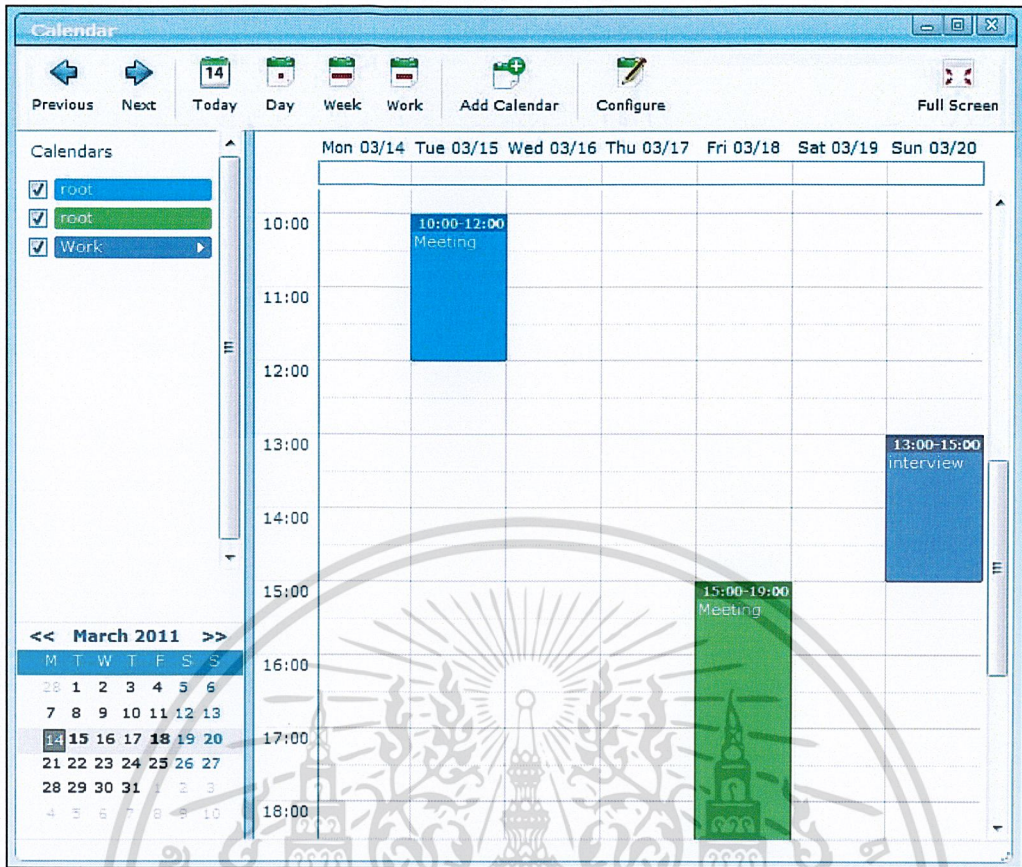
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เอฟทีพีไคลเอนต์ (FTP client)
- แอปพลิเคชันในการอ่าน RSS (RSS news reader) ดังรูปที่ 2.6

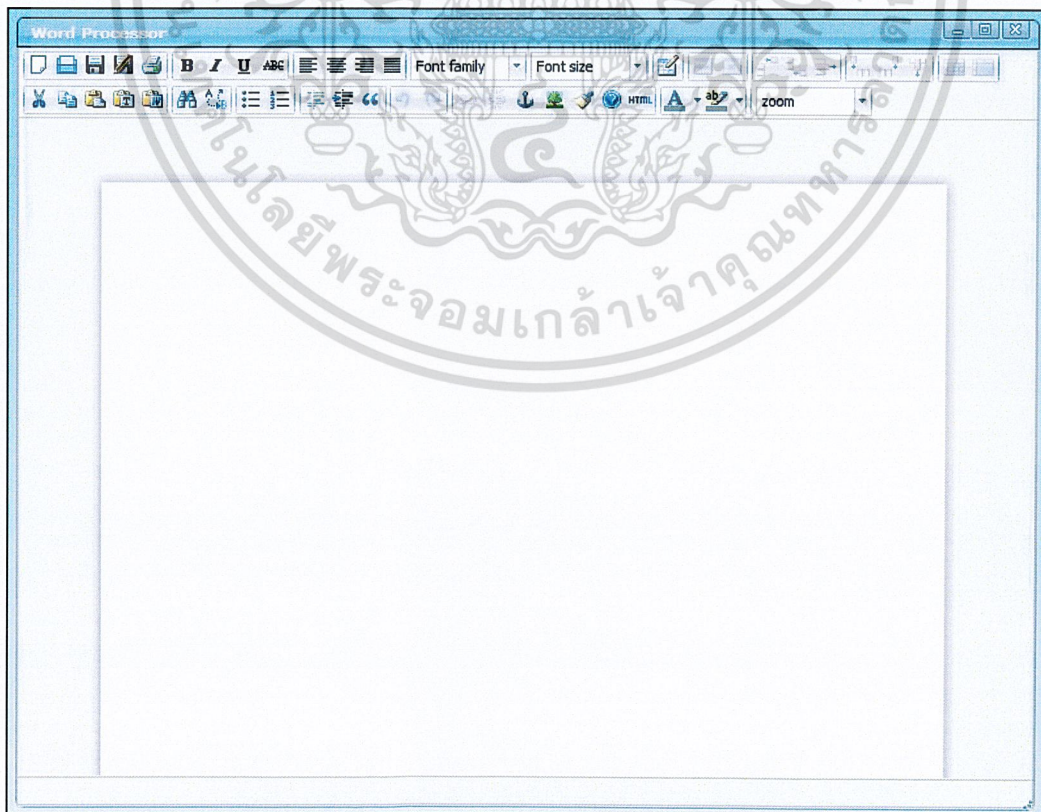


รูปที่ 2.3 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันสำหรับจัดการไฟล์ในอายุไอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

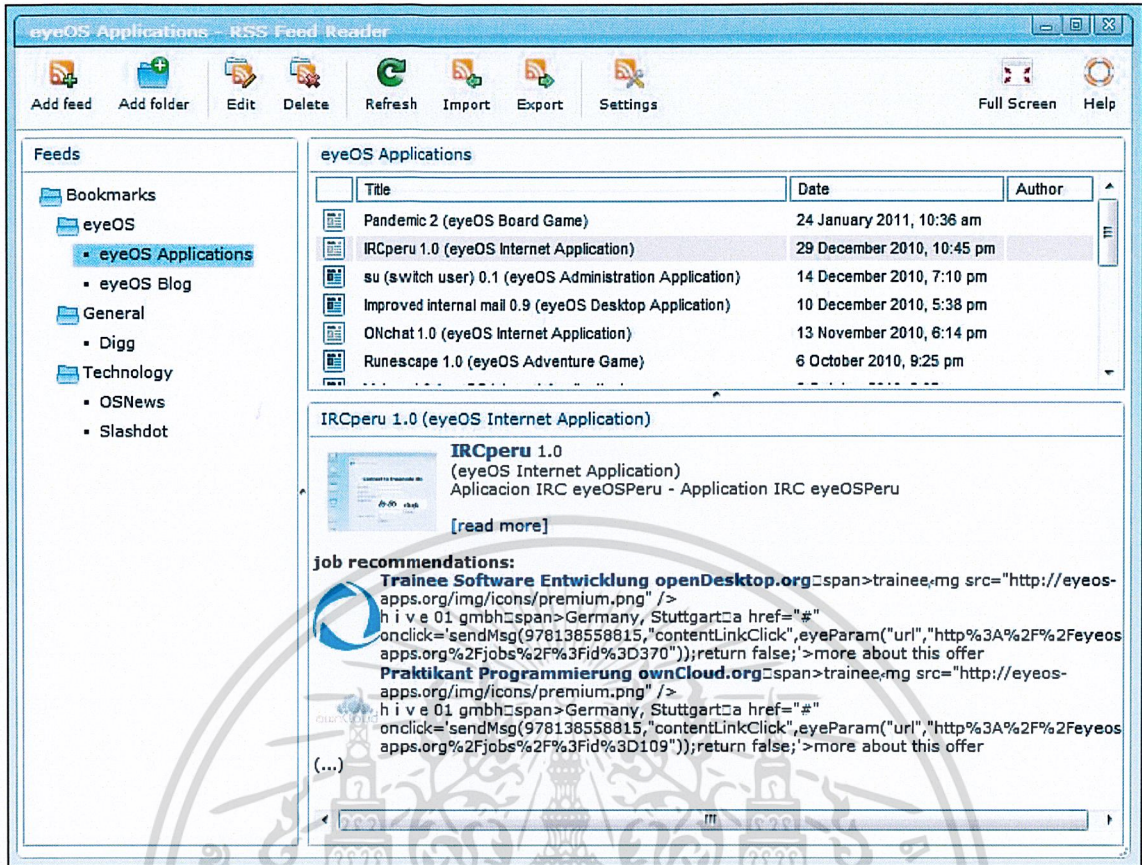


รูปที่ 2.4 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันปฏิทินส่วนตัวในอายุโอเอส



รูปที่ 2.5 ตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันประมวลผลคำในอายุโอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

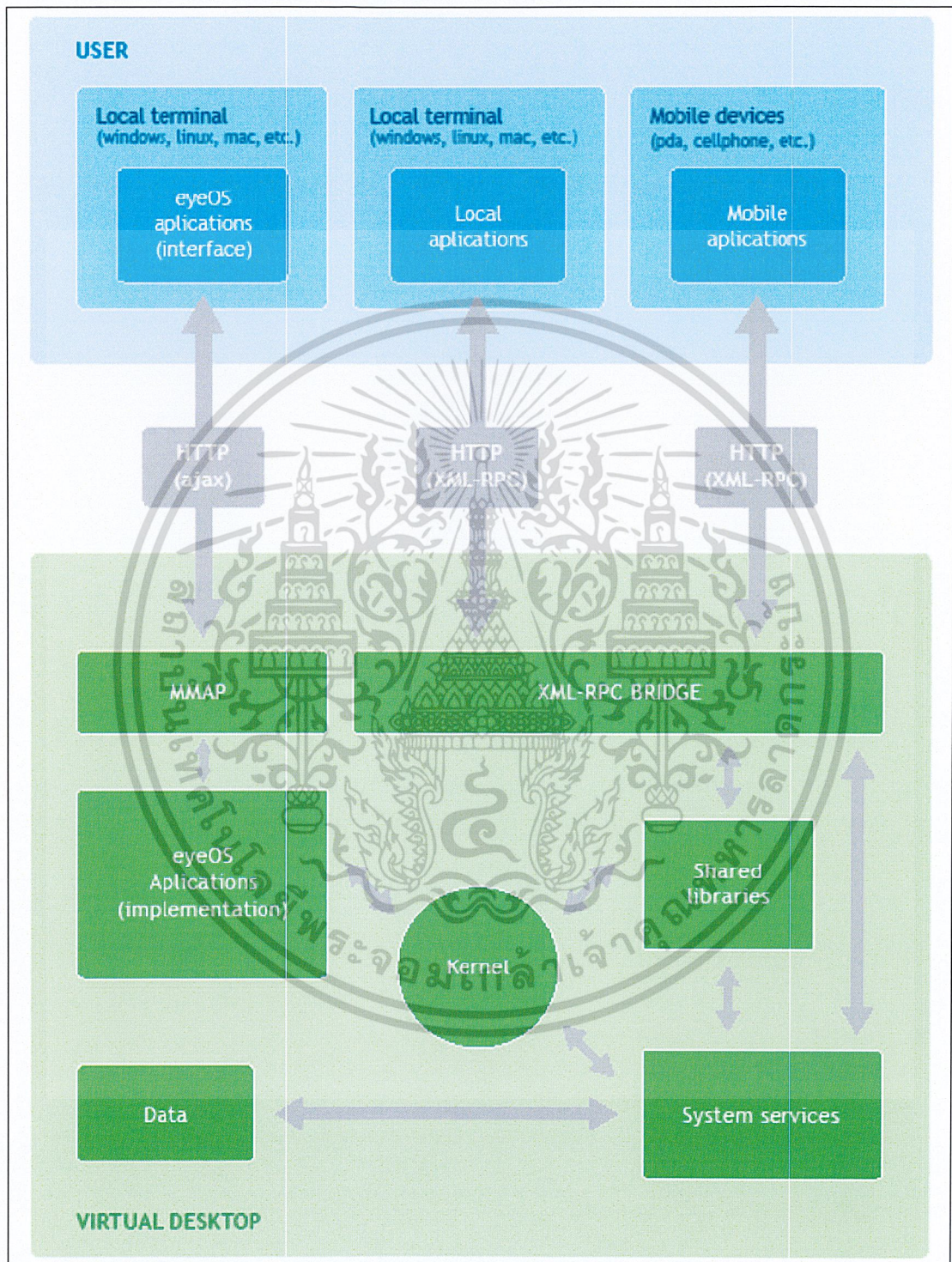


รูป 2.6 แสดงตัวอย่างส่วนติดต่อกับผู้ใช้ของแอปพลิเคชันสำหรับอ่าน RSS ในโรซ่าโอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1.2 สถาปัตยกรรมของอายโอเอส

สถาปัตยกรรมของอายโอเอสนั้น อธิบายได้ดังรูปที่ 2.7



รูปที่ 2.7 สถาปัตยกรรมของอายโอเอส (<http://www.eyeos.org/what-is-eyeos/technically/>)

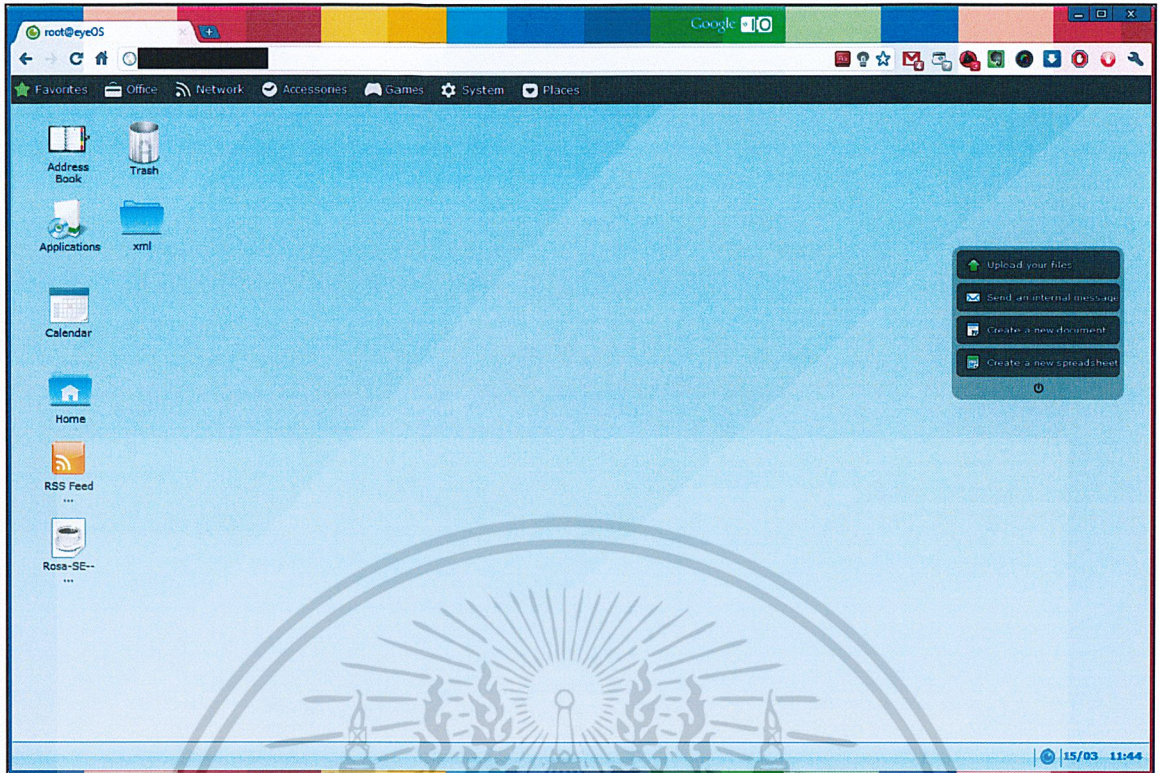
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.7 จะพบว่าอายุ โอเอส นั้นมีการแบ่งการทำงานออกเป็นสองส่วนด้วยกัน คือ ในส่วนที่เป็นไคลเอนต์ (จากรูปคือ User) และเซิร์ฟเวอร์ (จากรูปคือ Virtual Desktop) โดยไคลเอนต์จะเป็นอุปกรณ์คอมพิวเตอร์ที่มีระบบปฏิบัติการอยู่แล้ว โดยจากรูปสามารถ จำแนกการใช้งานออกเป็นสามรูปแบบดังนี้

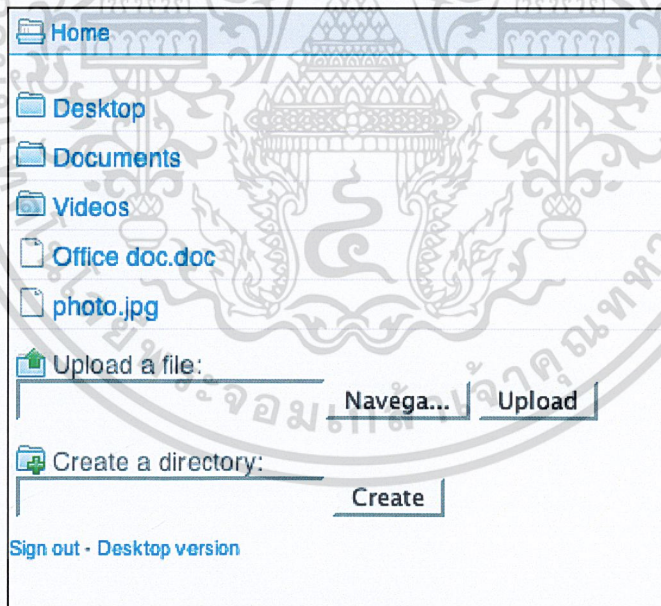
- 1) **ใช้งานผ่านส่วนติดต่อกับผู้ใช้ของอายุโอเอส (eyeOS applications (interface)):** โดยส่วนติดต่อกับผู้ใช้ของอายุโอเอสนั้นจะสามารถใช้งานผ่าน บราวเซอร์โดยเข้าไปยังเว็บไซต์ที่ติดตั้งอายุโอเอส ซึ่งในกรณีนี้บราวเซอร์ หรือไคลเอนต์จะติดต่อสื่อสารกับเซิร์ฟเวอร์ผ่านทางโปรโตคอล HTTP ด้วย เทคโนโลยี Ajax สามารถศึกษาตัวอย่างส่วนติดต่อกับผู้ใช้ของอายุโอเอสที่ใช้งานผ่านบราวเซอร์ได้จากรูปที่ 2.8
- 2) **ใช้งานผ่านแอปพลิเคชันที่ติดตั้งอยู่บนคอมพิวเตอร์ (Local applications):** แอปพลิเคชันที่สนับสนุนการทำงานร่วมกับอายุโอเอส เช่น OpenOffice นั้น จะสามารถติดต่อสื่อสารกับเซิร์ฟเวอร์ผ่านทางโปรโตคอล HTTP
- 3) **ใช้งานผ่านแอปพลิเคชันบนโทรศัพท์มือถือ (Mobile applications):** คือการใช้งานอายุโอเอสผ่านทางแอปพลิเคชันบนโทรศัพท์มือถือ รวมไปถึงการใช้งานผ่านทางบราวเซอร์บนโทรศัพท์มือถืออีกด้วย ซึ่งการติดต่อกับเซิร์ฟเวอร์นั้นจะติดต่อผ่านทางโปรโตคอล HTTP สามารถศึกษาตัวอย่างส่วนติดต่อกับผู้ใช้ของอายุโอเอสที่ใช้งานบนโทรศัพท์มือถือได้จากรูปที่ 2.9

เซิร์ฟเวอร์รับข้อมูลที่ถูกส่งมาจากไคลเอนต์ด้วย MMAP (Message MAPper) ที่ทำงานร่วมกับ XML-RPC Bridge ซึ่งข้อมูลที่ส่งไปและส่งกลับทั้งหมดนั้นจะอยู่ในรูปของ XML เสมอ หลังจากที่เซิร์ฟเวอร์ได้รับข้อมูลจะทำการประมวลผลตามที่ไคลเอนต์ทำการร้องขอ หลังจากประมวลผลเสร็จสิ้น เซิร์ฟเวอร์จะส่งผลลัพธ์ไปที่ MMAP จากนั้นผลลัพธ์จะถูกแปลงให้อยู่ในรูปแบบของ XML ก่อนถูกส่งไปที่ไคลเอนต์

การทำงานภายในเซิร์ฟเวอร์ของอายุโอเอสนั้น จะทำงานโดยมีเคอร์เนล (Kernel) เป็นตัวควบคุมการทำงานทั้งหมด ซึ่งมีการเรียกใช้งานไลบรารีส่วนกลาง (Shared libraries) เรียกใช้งานบริการของระบบ (System services) และจัดการแอปพลิเคชันของอายุโอเอส (eyeOS application) โดยในส่วนของการเข้าถึงข้อมูลไฟล์และข้อมูลของผู้ใช้ (Data) นั้น จะต้องเรียกใช้ผ่านทางเซิร์ฟวิสต์ทั้งหมด



รูป 2.8 ตัวอย่างส่วนติดต่อกับผู้ใช้ของอายโอเอสเมื่อใช้งานผ่านเบราว์เซอร์



รูปที่ 2.9 ตัวอย่างส่วนติดต่อกับผู้ใช้ของอายโอเอสเมื่อใช้งานผ่านโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

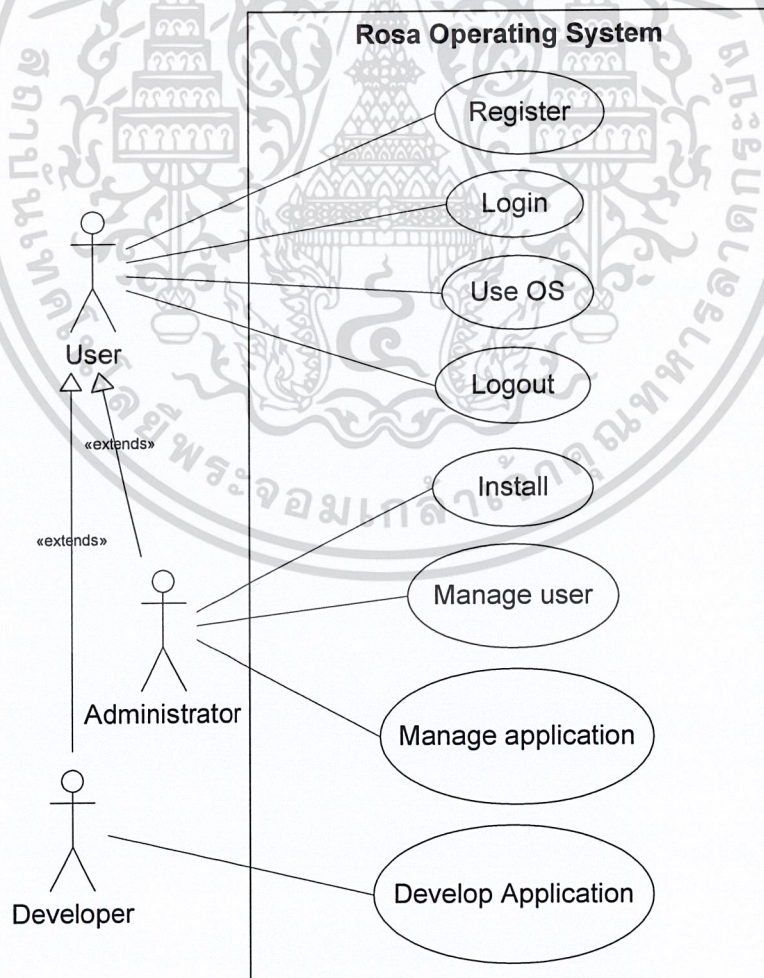
การวิเคราะห์และออกแบบระบบ

3.1 การวิเคราะห์ระบบ

ในขั้นตอนการวิเคราะห์ระบบของโรซ่าโอเอสนั้นจะแบ่งออกเป็นสองส่วนด้วยกันคือ ส่วนการวิเคราะห์เกี่ยวกับการใช้งานรวมถึงความสามารถของโรซ่าโอเอส และส่วนการวิเคราะห์โครงสร้างการทำงานของระบบซึ่งโครงสร้างนี้นั้นจะนำไปเป็นแนวคิดในการออกแบบสถาปัตยกรรม

3.1.1 วิเคราะห์การใช้งาน

โรซ่าโอเอสเป็นระบบปฏิบัติการบนเว็บไซต์ ที่สามารถใช้งานผ่านทางเว็บเบราว์เซอร์ของระบบปฏิบัติการใดก็ได้ที่มีการเชื่อมต่อกับอินเทอร์เน็ต ซึ่งโรซ่าโอเอสจะให้บริการผู้ใช้พร้อมกันมากกว่าหนึ่งคน โดยมีผู้ดูแลระบบที่คอยจัดการและควบคุมการใช้งานของผู้ใช้แต่ละคน รวมไปถึงนักพัฒนายังสามารถพัฒนาแอปพลิเคชันเพื่อทำงานบน โรซ่าโอเอส ได้ ดังรูปที่ 3.1



รูปที่ 3.1 ยูสเคสแสดงการใช้งานโรซ่าโอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 ผู้ใช้ (User)

ผู้ใช้นั้นจะเน้นไปที่การใช้งานเป็นหลัก สามารถจำแนกได้ดังนี้

- **ลงทะเบียนเข้าใช้งาน (Register):** ผู้ใช้จำเป็นต้องทำการลงทะเบียนเข้าใช้งานก่อนจึงจะสามารถใช้งานโรซ่าโอเอสได้
- **เข้าสู่ระบบ (Login):** หลังจากที่ผู้ใช้ทำการลงทะเบียนแล้วผู้ใช้งานจะสามารถเข้าสู่ระบบได้ผ่านการล็อกอิน โดยผู้ใช้ที่ไม่ได้ลงทะเบียนจะไม่สามารถล็อกอินเข้าใช้งานได้
- **ใช้งานระบบ (Use OS):** หลังจากที่ผู้ใช้ได้ล็อกอินเข้าสู่ระบบแล้ว ผู้ใช้งานจะสามารถใช้งานโรซ่าโอเอสได้ โดยผู้ใช้งานจะสามารถจัดการไฟล์ของตนเอง รวมไปถึงใช้งานแอปพลิเคชันที่ระบบจัดเตรียมไว้ให้ได้
- **ออกจากระบบ (Logout):** เมื่อผู้ใช้เสร็จสิ้นการใช้งาน ผู้ใช้สามารถออกจากระบบได้

3.1.1.2 ผู้ดูแลระบบ (Administrator)

ผู้ดูแลระบบสามารถใช้งานระบบได้เหมือนกับผู้ใช้ทั่วไป แต่สามารถจัดการและควบคุมการใช้งานของผู้ใช้ได้ ดังนี้

- **จัดการผู้ใช้ (Manage User):** สามารถจัดการผู้ใช้ได้ เช่น การเรียกดูข้อมูลของผู้ใช้ การลบผู้ใช้ และการแก้ไขข้อมูลของผู้ใช้
- **จัดการแอปพลิเคชัน (Manage Application):** สามารถจัดการแอปพลิเคชันได้ เช่น เพิ่มแอปพลิเคชัน ลบแอปพลิเคชัน และเรียกดูรายละเอียดของแอปพลิเคชัน
- **การติดตั้ง (Install):** สามารถติดตั้งโรซ่าโอเอสได้เมื่อมีการใช้งานครั้งแรก โดยระบบจะมีโปรแกรมสำหรับช่วยเหลือในการติดตั้งให้

3.1.1.3 นักพัฒนา (Developer)

นักพัฒนานั้นสามารถพัฒนาแอปพลิเคชันและทำการทดสอบบนโรซ่าโอเอสได้

- **พัฒนาแอปพลิเคชัน (Develop Application):** สามารถพัฒนาแอปพลิเคชันให้ทำงานกับโรซ่าโอเอสได้ โดยโรซ่าโอเอสจะมีการสนับสนุนในด้านต่างๆ ให้

3.1.2 วิเคราะห์โครงสร้างการทำงาน

สถาปัตยกรรมของ โรซ่าโอเอส นั้นได้อ้างอิงมาจากสถาปัตยกรรมของอายโอเอส (eyeOS) ที่กล่าวไว้ในบทที่สอง โดยมีการแบ่งส่วนของการทำงานออกเป็นสองส่วนคือ ส่วนที่เป็นไคลเอนต์ และส่วนที่เป็นเซิร์ฟเวอร์

3.1.2.1 ไคลเอนต์ (Client)

มีหน้าที่ในการจัดการเกี่ยวกับส่วนติดต่อกับผู้ใช้และแอปพลิเคชัน ซึ่งมีคุณสมบัติดังต่อไปนี้

- ส่วนติดต่อกับผู้ใช้มีความสอดคล้องกันทั้งระบบ เพื่อลดความสับสนในการใช้งาน
- ผู้ใช้สามารถควบคุมการทำงานของแอปพลิเคชันได้อย่างสมบูรณ์ เช่น การเปิดและปิดแอปพลิเคชัน
- ระบบสามารถตรวจสอบประเภทของไฟล์เพื่อเรียกใช้งานแอปพลิเคชันที่เหมาะสมโดยอัตโนมัติเมื่อผู้ใช้สั่งเปิดไฟล์

แอปพลิเคชันสามารถแลกเปลี่ยนข้อมูลกันในขณะที่ทำงานได้ เพื่อสนับสนุนการทำงานร่วมกันของแอปพลิเคชัน

3.1.2.2 เซิร์ฟเวอร์ (Server)

มีหน้าที่ในการจัดการระบบไฟล์ ระบบแอปพลิเคชัน ระบบผู้ใช้งาน และระบบของผู้ดูแลระบบ ซึ่งมีความสามารถดังต่อไปนี้

- ระบบจัดการไฟล์สามารถจัดการไฟล์ขั้นพื้นฐานได้ ซึ่งจะอธิบายเพิ่มเติมในหัวข้อถัดไป
- ระบบสำหรับผู้ดูแลระบบสามารถจัดการข้อมูลเกี่ยวกับผู้ใช้ เช่น การแก้ไขข้อมูล และลบผู้ใช้ รวมถึงสามารถจัดการแอปพลิเคชันของผู้ใช้ เช่น การเพิ่มแอปพลิเคชัน ลบแอปพลิเคชัน และดูรายละเอียดของแต่ละแอปพลิเคชันได้
- ระบบจัดการแอปพลิเคชันที่สามารถทำงานร่วมกับแอปพลิเคชันที่มีอยู่ในระบบได้ รวมถึงรองรับการเพิ่มแอปพลิเคชันใหม่ที่นักพัฒนาได้พัฒนาขึ้นมาเพิ่มเติมด้วย

ระบบที่รองรับการร้องขอข้อมูลจากไคลเอนต์ เพื่อให้แอปพลิเคชันสามารถทำงาน บางส่วนบนเซิร์ฟเวอร์ได้

3.1.3 วิเคราะห์ระบบไฟล์

ระบบไฟล์ของระบบปฏิบัติการที่ติดตั้งนั้นจะมีคุณสมบัติที่สำคัญคือ ความปลอดภัย ความรวดเร็ว และสามารถจัดการไฟล์ขั้นพื้นฐาน เช่น เขียนไฟล์ อ่านไฟล์ สร้างไฟล์ ลบไฟล์ เปลี่ยนชื่อไฟล์ เป็นต้น ซึ่งระบบไฟล์ถือเป็นส่วนสำคัญของโรซ่า โอเอส เนื่องการทำงานหลักของแอปพลิเคชันส่วนใหญ่จะมีความเกี่ยวข้องกับไฟล์ และเป็นระบบที่มีผู้ใช้พร้อมกันครั้งละหลายคน ดังนั้น เพื่อให้ไฟล์ของผู้ใช้แต่ละคนมีความปลอดภัย ความรวดเร็ว และสามารถรองรับการที่มีผู้ใช้พร้อมกันครั้งละหลายคนได้ ระบบไฟล์จึงมีคุณสมบัติดังต่อไปนี้

3.1.3.1 ความปลอดภัย

เพื่อความปลอดภัยของผู้ใช้ ระบบไฟล์จะอนุญาตให้เจ้าของไฟล์แต่เพียงผู้เดียวเท่านั้น ที่สามารถจัดการไฟล์ของตนเองได้ ผู้ที่ไม่ใช่เจ้าของไฟล์จะไม่สามารถค้นหาและทราบที่อยู่ของไฟล์ของผู้อื่น ไฟล์จะไม่มีการสูญหายไปจากระบบถ้าเจ้าของไม่ได้ทำการลบไฟล์ดังกล่าว เว้นแต่หากบัญชีผู้ใช้ถูกลบออกจากระบบ ไฟล์ของผู้ใช้ก็จะถูกลบออกจากระบบทั้งหมดในทันที

3.1.3.2 ความรวดเร็วในการรวบรวมข้อมูล

ระบบไฟล์สามารถรวบรวมข้อมูลที่เกี่ยวข้องกับไฟล์ได้รวดเร็ว ทำให้การใช้งานมีความรวดเร็ว โดยตัวอย่างการรวบรวมข้อมูลเช่น การค้นหาขนาดของไฟล์ทั้งหมดที่ผู้ใช้คนหนึ่งเป็นเจ้าของ การค้นหาขนาดของโพลีเคอร์และไฟล์ รวมไปถึงการค้นหาข้อมูลไฟล์จากคำค้นที่ต้องการ

3.1.3.3 ความสะดวกในการเข้าถึง

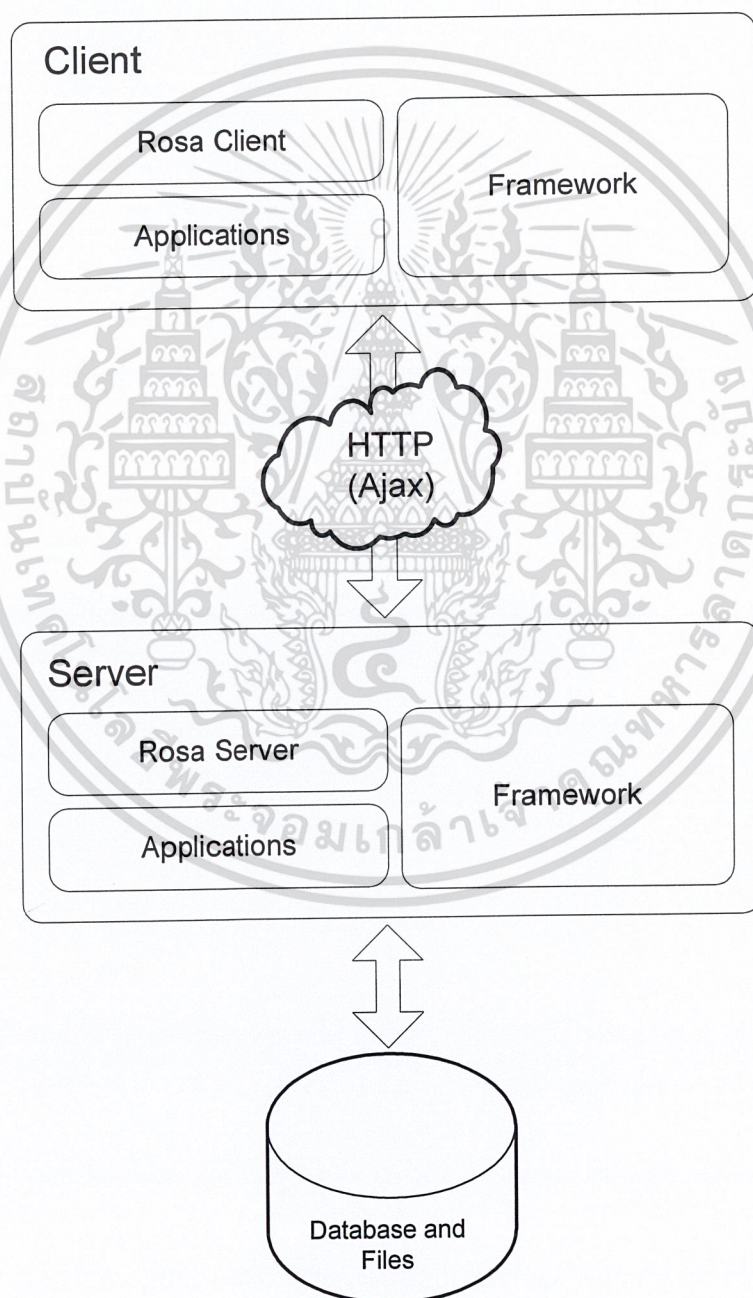
ระบบไฟล์สามารถเข้าถึงและใช้งานไฟล์ได้อย่างสะดวก โดยสามารถเข้าถึงไฟล์โดยตรงผ่านทาง URL เพื่อให้แอปพลิเคชันที่จำเป็นต้องใช้ URL ในการเปิดไฟล์สามารถนำไปใช้งานได้ทันที ซึ่ง URL ที่ใช้เข้าถึงไฟล์จะมีเพียงระบบไฟล์และเจ้าของไฟล์เท่านั้นที่ทราบ

3.2 การออกแบบระบบ

3.2.1 สถาปัตยกรรม

สถาปัตยกรรมของ โรซ่าโอเอส นั้นแบ่งออกเป็นสองส่วนคือ ส่วนที่เป็นไคลเอนต์ และส่วนที่เป็นเซิร์ฟเวอร์ ซึ่งในส่วนไคลเอนต์นั้นจะมีหน้าที่ในการจัดการส่วนติดต่อกับผู้ใช้ และส่วนที่เป็นเซิร์ฟเวอร์จะทำหน้าที่ในการจัดการไฟล์ จัดการผู้ใช้ จัดการแอปพลิเคชัน และเป็นส่วนของผู้ดูแลระบบ โดยทั้งสองส่วนนั้นติดต่อสื่อสารกันผ่านโปรโตคอล HTTP ด้วยเทคโนโลยี Ajax ดังรูปที่

3.2

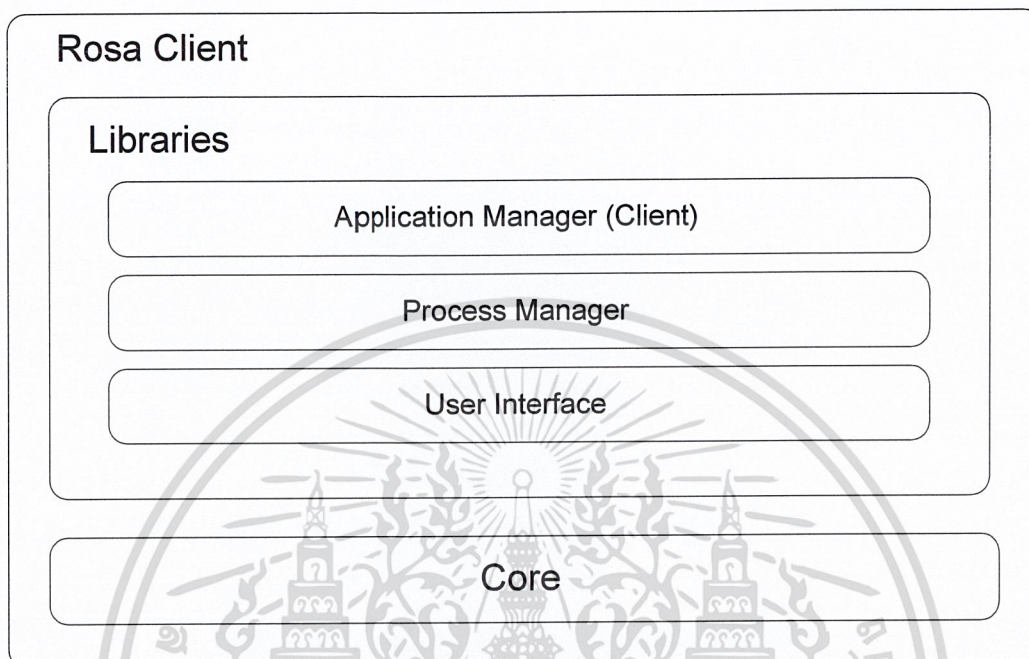


รูปที่ 3.2 สถาปัตยกรรมของโรซ่าโอเอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.1 โรซ่าไคลเอนต์ (Rosa Client)

จากการวิเคราะห์โรซ่าไคลเอนต์มีหน้าที่ในการจัดการส่วนติดต่อกับผู้ใช้และแอปพลิเคชัน จึงสามารถออกแบบโรซ่าไคลเอนต์ได้ดังรูปที่ 3.3



รูปที่ 3.3 โครงสร้างระบบโรซ่าไคลเอนต์ทางฝั่งไคลเอนต์

จะพบว่ามีการแบ่งออกเป็นสองส่วนด้วยกันคือ ส่วนควบคุมการทำงานหลัก (Core) และส่วนของไลบรารี (Libraries) ซึ่งส่วนควบคุมการทำงานหลักนั้นจะมีการทำงานอัตโนมัติ และคอยควบคุมการทำงานของไลบรารี

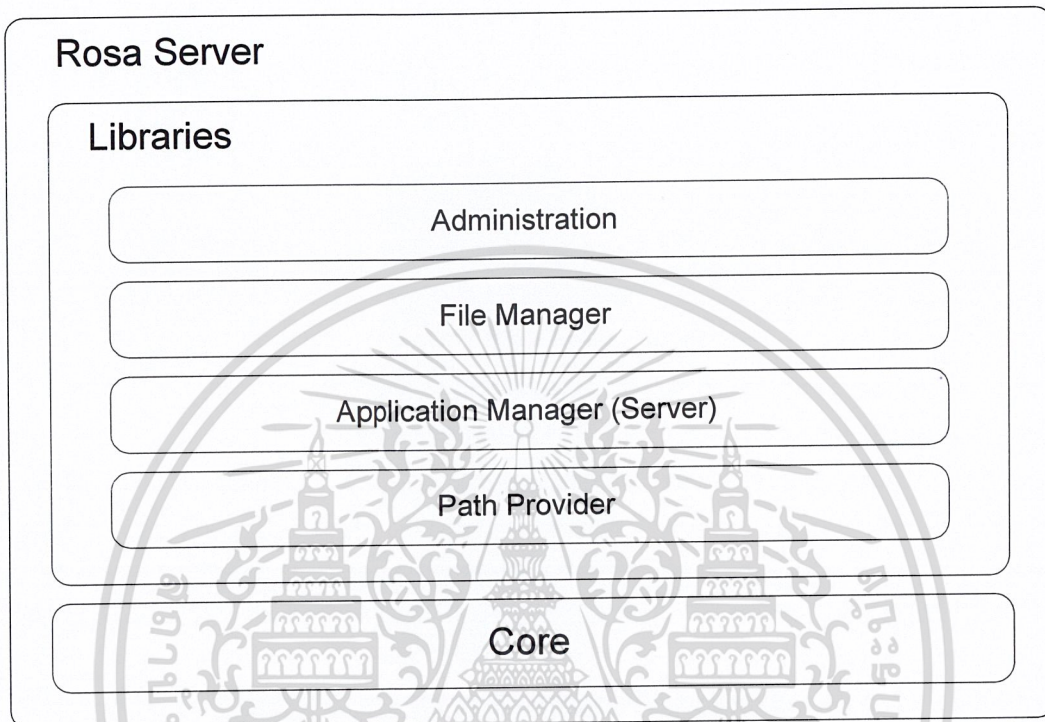
โรซ่าไคลเอนต์ไลบรารี (Rosa Client Libraries) สามารถแบ่งออกเป็น 3 ส่วน

- 1) ส่วนจัดการแอปพลิเคชัน (Application Manager (Client)): มีหน้าที่ในการจัดการแอปพลิเคชันภายในโรซ่าไคลเอนต์ โดยจะจัดการเกี่ยวกับส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน และการเรียกใช้งานแอปพลิเคชันภายในระบบ
- 2) ส่วนจัดการโปรเซส (Process Manager): มีหน้าที่ในการจัดการโปรเซส ตั้งแต่การสร้างโปรเซส การทำลายโปรเซส และการส่งสัญญาณระหว่างโปรเซส
- 3) ส่วนติดต่อกับผู้ใช้ (User Interface): เป็นส่วนเพิ่มเติมสำหรับการพัฒนาส่วนติดต่อกับผู้ใช้ เพื่อตอบสนองเว็บแอปพลิเคชันที่มีส่วนติดต่อกับผู้ใช้

3.2.1.2 โรซ่าเซิร์ฟเวอร์ (Rosa Server)

จากการวิเคราะห์นั้นโรซ่าเซิร์ฟเวอร์มีหน้าที่ในการจัดการระบบไฟล์ ระบบแอปพลิเคชัน ระบบผู้ใช้งาน และระบบของผู้ดูแลระบบ จึงสามารถออกแบบ โรซ่าเซิร์ฟเวอร์ได้ดังรูปที่

3.4



รูปที่ 3.4 โครงสร้างระบบโรซ่าโอเอสทางฝั่งเซิร์ฟเวอร์

จะพบว่ามีลักษณะโครงสร้างคล้ายโรซ่าไคลเอนต์ คือมีส่วนควบคุมการทำงานหลัก (Core) และไลบรารี (Libraries) ซึ่งส่วนควบคุมการทำงานหลักนั้นจะมีการทำงานอัตโนมัติ และคอยควบคุมการทำงานของไลบรารีนั่นเอง

โรซ่าเซิร์ฟเวอร์ไลบรารี (Rosa Server Libraries) สามารถแบ่งออกเป็น 4 ส่วน

- 1) ส่วนดูแลจัดการระบบ (Administration): มีหน้าที่ในการจัดการข้อมูลของผู้ใช้และแอปพลิเคชันต่างๆของผู้ใช้ทุกคน
- 2) ส่วนจัดการไฟล์ (File System): มีหน้าที่ในการจัดการเกี่ยวกับระบบไฟล์ เช่น การสร้างไฟล์ ลบไฟล์ เปลี่ยนชื่อไฟล์ การอัปโหลด เป็นต้น
- 3) ส่วนจัดการแอปพลิเคชัน (Application Manager (Server)): มีหน้าที่ในการจัดการแอปพลิเคชันภายในโรซ่าโอเอส โดยจะจัดการเกี่ยวกับการนำไฟล์ของ

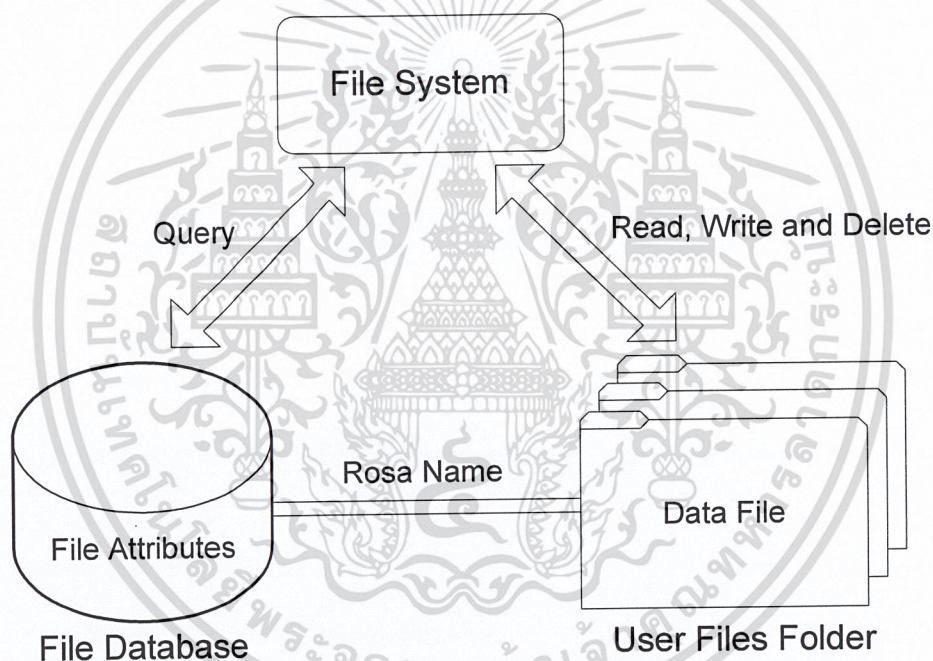
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอปพลิเคชันไปทำงานในไคลเอนต์ และจัดการเกี่ยวกับการประมวลผลการร้องขอแอปพลิเคชันบนเซิร์ฟเวอร์ที่แอปพลิเคชันทางไคลเอนต์ได้ร้องขอมา

- 4) ส่วนให้บริการที่อยู่ของไฟล์ (Path Provider): เป็นส่วนที่ให้บริการในการเปลี่ยนแปลงที่อยู่ของไฟล์ (Path) ที่ได้รับจากทางผู้เข้ามาเป็นที่อยู่จริงของไฟล์ในเซิร์ฟเวอร์

3.2.2 การออกแบบระบบไฟล์

จากการวิเคราะห์ระบบไฟล์ ระบบไฟล์ของโรซ่าโอเอสมีคุณสมบัติสำคัญคือ การรักษาความปลอดภัย ความรวดเร็วในการรวบรวมข้อมูล และความสะดวกในการเข้าถึง จึงสามารถออกแบบระบบไฟล์ได้ดังนี้



รูป 3.5 ระบบไฟล์ของโรซ่าโอเอส

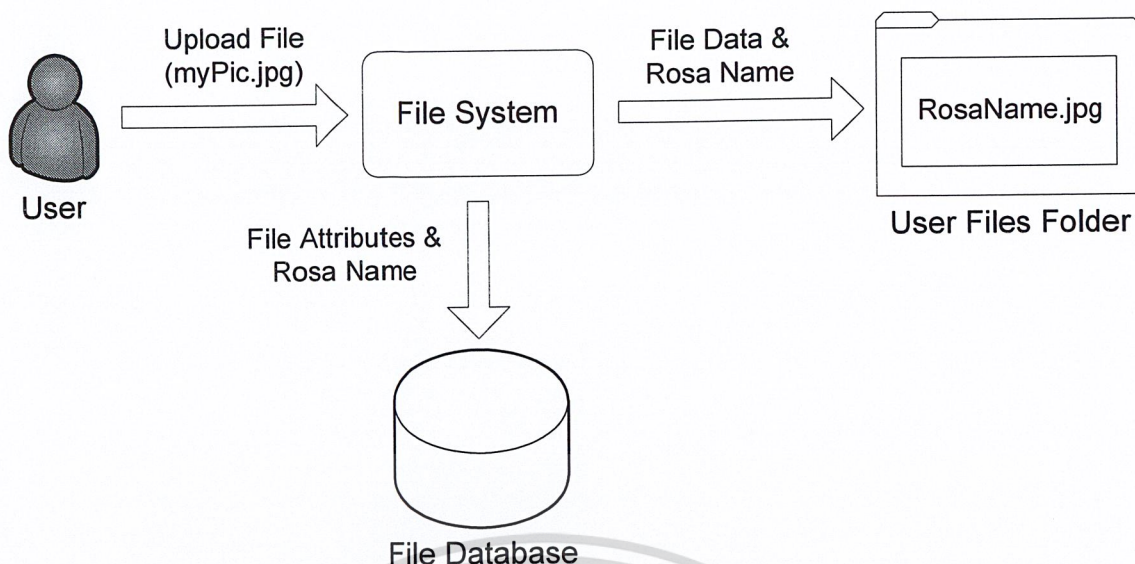
จากรูปที่ 3.5 จะพบว่า ระบบไฟล์ของโรซ่าโอเอสนั้นมีการนำฐานข้อมูลมาใช้งาน โดยในที่นี้จะเรียกว่า ฐานข้อมูลไฟล์ (File Database) ซึ่งใช้สำหรับจัดเก็บลักษณะประจำไฟล์ (File Attribute) เช่น รหัสประจำไฟล์ ชื่อไฟล์ ขนาดไฟล์ ประเภทไฟล์ และเจ้าของไฟล์ เป็นต้น โดยไฟล์ข้อมูล (Data File) นั้นจะถูกจัดเก็บรวมกันไว้ที่โพลเดอร์หนึ่งในที่นี้จะเรียกว่า โพลเดอร์ยูสเซอร์ไฟล์ (User Files Folder) ซึ่งไฟล์ข้อมูลและลักษณะประจำไฟล์นั้น สามารถอ้างอิงถึงกันผ่านชื่อกลางหรือในที่นี้จะเรียกว่า โรซ่าเนม (Rosa name) ซึ่งโรซ่าเนมจะถูกนำไปตั้งเป็นชื่อไฟล์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และถูกจัดเก็บอยู่ในลักษณะประจำไฟล์ด้วย โดยฐานข้อมูลไฟล์และโพลเดอรัยสเซอร์ไฟล์นั้นจะสามารถเข้าถึงและใช้งานโดยส่วนจัดการไฟล์ (File System) ซึ่งส่วนจัดการไฟล์จะเป็นตัวกลางในการเข้าถึงเพื่อจัดการลักษณะประจำไฟล์ และไฟล์ข้อมูล ซึ่งในที่นี้หมายถึง การอ่านไฟล์ การเขียนไฟล์ การลบไฟล์ การสร้างไฟล์ การอัปโหลดไฟล์ การเปลี่ยนชื่อไฟล์ เป็นต้น จากการออกแบบทำให้ได้ระบบไฟล์ที่มีคุณสมบัติตามที่ต้องการ สามารถอธิบายได้ดังนี้

- **ความปลอดภัย:** ระบบจัดการไฟล์จะอนุญาตให้เจ้าของไฟล์สามารถเข้าถึงลักษณะประจำไฟล์ได้เท่านั้น ทำให้โรซ่าเนมนั้นถูกเก็บเป็นความลับ โดยระบบไฟล์จะตรวจสอบความเป็นเจ้าของผ่านทางรายละเอียดของไฟล์
- **ความเร็วในการรวบรวมข้อมูล:** เนื่องจากมีการนำเอาฐานข้อมูลเข้ามาใช้งาน จึงทำให้สามารถรวบรวมข้อมูล และค้นหาไฟล์ได้อย่างรวดเร็ว
- **ความสะดวกในการใช้งานไฟล์:** เนื่องจากระบบไฟล์มีการจัดเก็บไฟล์ข้อมูลให้สามารถเข้าถึงได้โดยตรง หรือผ่านทาง URL (หากทราบโรซ่าเนม) ทำให้แอปพลิเคชันทางฝั่งไคลเอนต์สามารถนำไฟล์ออกมาใช้งานได้ผ่านทาง URL และแอปพลิเคชันบนเซิร์ฟเวอร์สามารถเข้าถึงไฟล์ที่อยู่ในยูสเซอร์ไฟล์ได้โดยตรง

รูปที่ 3.6 แสดงขั้นตอนการทำงานของระบบไฟล์เมื่อผู้ใช้อัปโหลดไฟล์ myPic.jpg เข้าสู่ระบบ ในขั้นตอนแรกระบบไฟล์จะวิเคราะห์ไฟล์ข้อมูลเพื่อหาลักษณะประจำไฟล์รวมถึงทำการสุ่มโรซ่าเนม จากนั้นระบบจะนำลักษณะประจำไฟล์และโรซ่าเนมไปจัดเก็บไว้ที่ฐานข้อมูลไฟล์ ต่อมาระบบจะนำไฟล์ข้อมูลไปเก็บไว้ที่โพลเดอรัยสเซอร์ไฟล์ โดยไฟล์ข้อมูลนั้นจะถูกเปลี่ยนชื่อให้เหมือนกับโรซ่าเนม



รูปที่ 3.6 ตัวอย่างการทำงานของระบบไฟล์ เมื่อผู้ใช้อัพโหลดไฟล์เข้าสู่ระบบ

3.2.3 การออกแบบส่วนติดต่อกับผู้ใช้

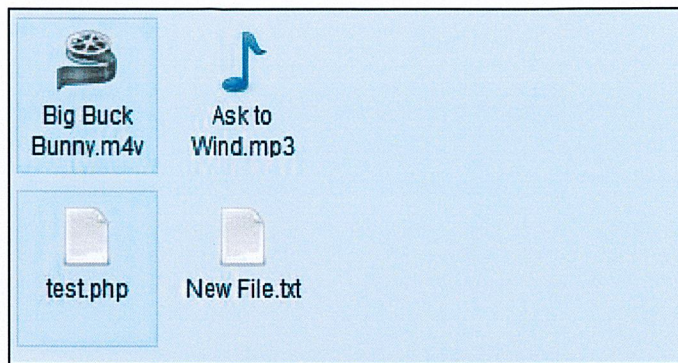
ส่วนติดต่อกับผู้ใช้ของโรซ่าโอเอสนั้นมีการอ้างอิงรูปแบบมาจากระบบปฏิบัติการ Windows XP โดยรูปแบบที่อ้างอิงมานั้น มีทั้งหมด 5 ส่วนด้วยกัน คือ

1. เดสก์ทอป (Desktop)
2. ป๊อปอัพเมนู (Popup menu หรือ Right-click-menu)
3. หน้าต่าง (Window)
4. ทาสก์บาร์ (Taskbar)
5. สตาร์ทเมนู (Start menu)

ซึ่งทั้งหมดที่กล่าวมานั้นถูกออกแบบให้มีความใกล้เคียงกับส่วนติดต่อกับผู้ใช้ของ Windows XP มากที่สุดเท่าที่จะเป็นไปได้ เพื่อให้ผู้ใช้เกิดความคุ้นเคยและเรียนรู้การใช้งานโรซ่าโอเอสได้อย่างรวดเร็วที่สุด

3.2.3.1 เดสก์ทอป (Desktop)

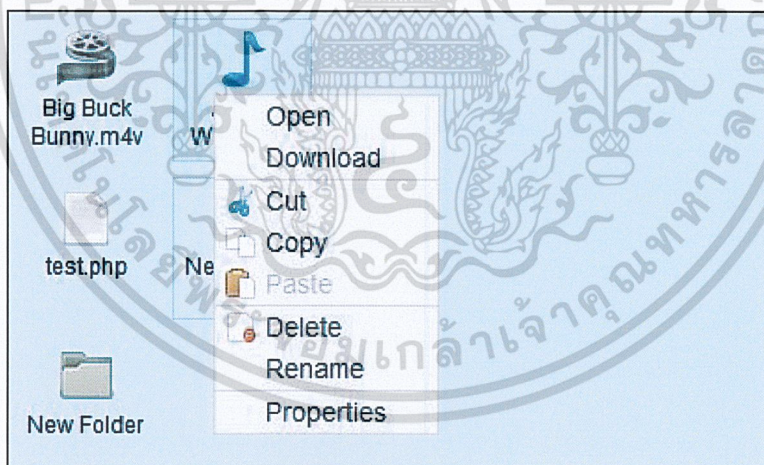
เดสก์ทอปของโรซ่าโอเอสนั้นมีความสามารถในการทำงานคือ ผู้ใช้สามารถคลิกเมาส์ขวาเพื่อแสดงป๊อปอัพเมนู สามารถแสดงไฟล์ที่อยู่บนเดสก์ทอป สามารถจัดการไฟล์ที่อยู่บนเดสก์ทอป (สร้าง แก้ไข และลบ) และสามารถเลือก (Select) ไฟล์ได้ โดยสามารถเลือกได้ครั้งละหนึ่งไฟล์หรือมากกว่า แสดงได้ดังรูปที่ 3.7



รูปที่ 3.7 การเลือกไฟล์มากกว่าหนึ่งไฟล์บนเดสก์ท็อป

3.2.3.2 ป๊อปอัพเมนู (Popup menu หรือ Right-click menu)

ป๊อปอัพเมนูหรือเมนูที่แสดงเมื่อผู้ใช้คลิกเมาส์ขวา เป็นเมนูที่ผู้ใช้สามารถสั่งให้แสดงและซ่อนได้ทุกเมื่อที่ต้องการ โดยข้อดีของเมนูในรูปแบบนี้คือระบบจะไม่แสดงเมนูนี้ไว้ตลอดเวลา แต่จะแสดงต่อเมื่อผู้ใช้คลิกเมาส์ขวาเท่านั้น จึงทำให้ผู้ใช้มีเนื้อที่การใช้งานบนหน้าจอเพิ่มขึ้น นอกจากนี้ยังทำให้เกิดความสอดคล้องในใช้งาน เนื่องจากเมนูที่ปรากฏขึ้นจากการคลิกเมาส์ขวาลงบนวัตถุใดๆ นั้นเป็นเมนูที่มีความสัมพันธ์กับวัตถุนั้นๆ โดยตรง ลักษณะของป๊อปอัพเมนูสามารถแสดงให้เห็นได้ ดังรูปที่ 3.8

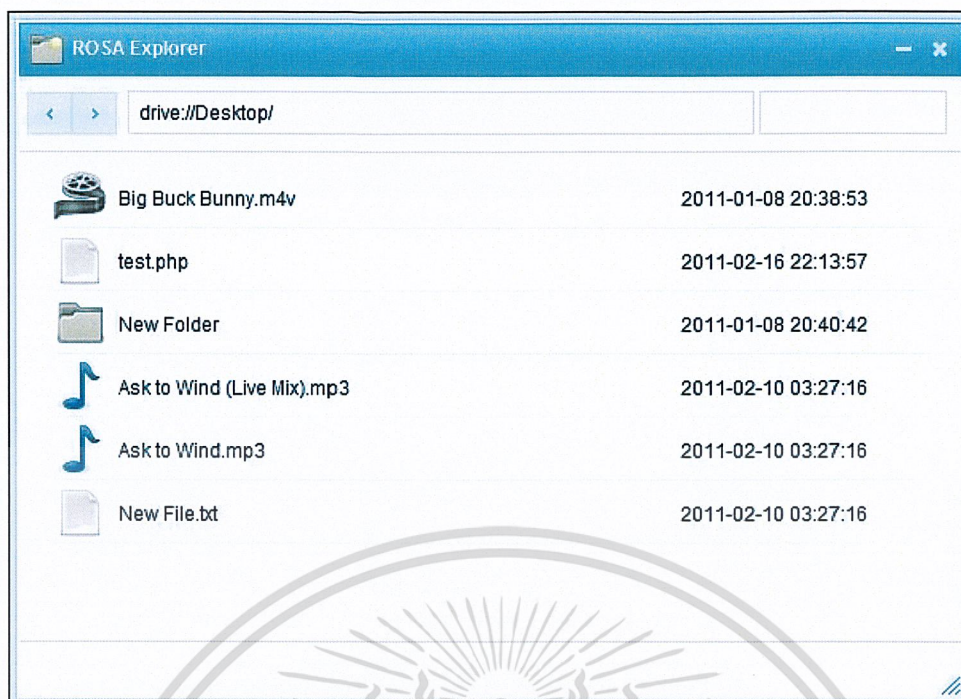


รูปที่ 3.8 ป๊อปอัพเมนู

3.2.3.3 หน้าต่าง (Window)

หน้าต่างมีลักษณะเป็นกรอบสี่เหลี่ยมที่แสดงส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน โดยที่มุมขวาบนของหน้าต่างภายใน โรซ่าโอเอส นั้นจะแสดงปุ่มปิดและปุ่มย่อหน้าต่าง โดยหน้าต่างจะทำงานร่วมกับทาสก์บาร์ ดังรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 หน้าต่างภายในโรซ่าโอเอส

3.2.3.4 ทาสก์บาร์ (Taskbar)

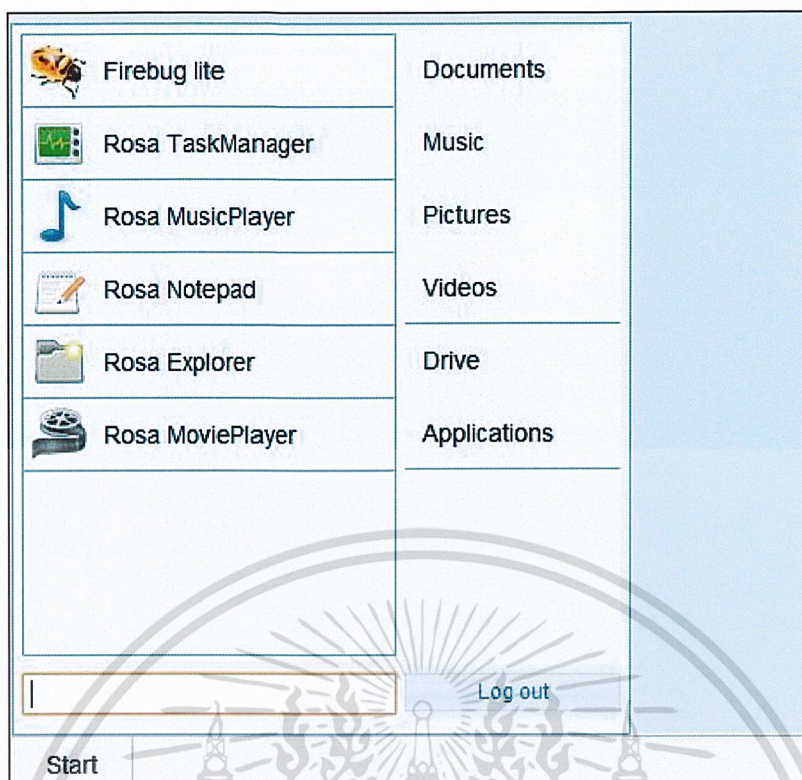
ทาสก์บาร์เป็นส่วนที่มีไว้สำหรับให้ผู้ใช้สลับการใช้งาน ไปมาระหว่างแอปพลิเคชันต่างๆ ได้ นอกจากนี้ยังใช้แสดงว่าในขณะนั้นมีหน้าต่างของแอปพลิเคชันใดบ้างที่แสดงและซ่อนอยู่ด้วย โดยผู้ใช้สามารถสั่งให้แสดงและซ่อนหน้าต่างของแอปพลิเคชันต่างๆ ได้จากทาสก์บาร์ ดังรูปที่ 3.10



รูปที่ 3.10 ทาสก์บาร์

3.2.3.5 สตาร์ทเมนู (Start menu)

สตาร์ทเมนูเป็นส่วนที่มีการรวบรวมแอปพลิเคชันทั้งหมดที่ติดตั้งไว้ภายในระบบ มาจัดแสดงเพื่อเตรียมพร้อมสำหรับการเรียกใช้งาน โดยมีการเพิ่มปุ่มที่ใช้ในการลือกเอาที่ออกจากระบบ และส่วนที่ใช้ในการเปิดโพลเดอร์พื้นฐานต่างๆ ของระบบเข้ามาด้วย ซึ่งทั้งสองส่วนที่เพิ่มเข้ามานั้นจะจัดแสดงอยู่บริเวณทางขวามือของสตาร์ทเมนู ดังรูปที่ 3.10



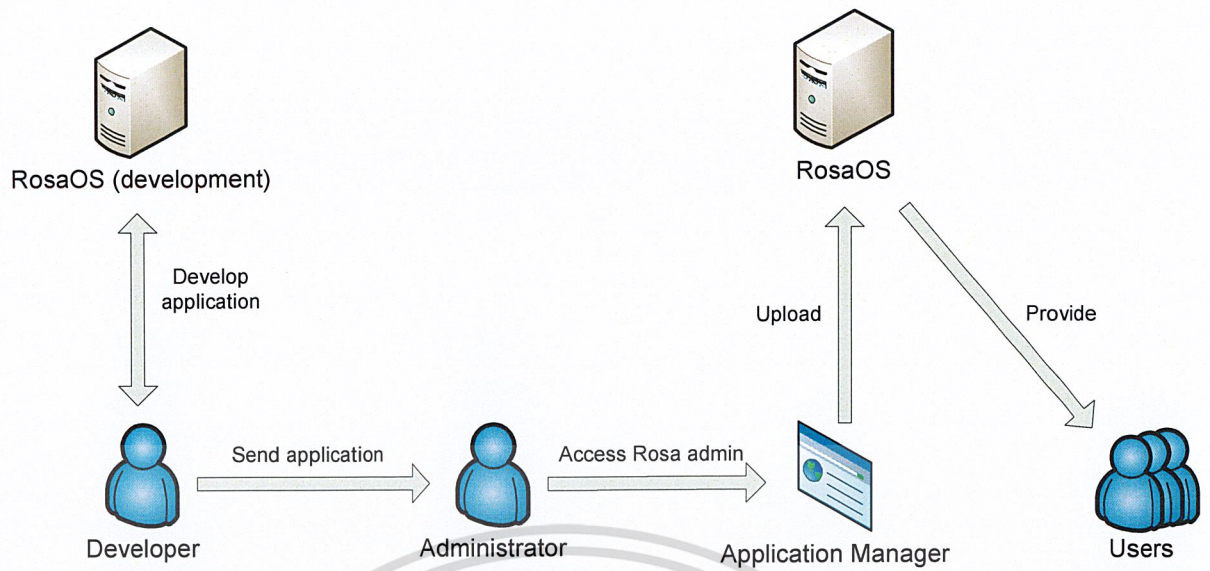
รูปที่ 3.11 สตาร์ทเมนู

3.2.4 การออกแบบขั้นตอนในการพัฒนาและการนำแอปพลิเคชันไปใช้งาน

การพัฒนาแอปพลิเคชันและการนำแอปพลิเคชันไปใช้งานบนโรซ่าโอเอสนั้น เพื่อให้เกิดความสะดวกในการติดตั้งและถอนแอปพลิเคชันออกจากระบบ ความสะดวกในการจัดเก็บแอปพลิเคชัน รวมไปถึงความสะดวกในการนำแอปพลิเคชันไปเผยแพร่ จึงสามารถออกแบบได้ดังนี้ (สามารถศึกษาภาพประกอบคำอธิบายได้จากรูปที่ 3.12)

- 1) นักพัฒนาทำการพัฒนาแอปพลิเคชันลงบน โรซ่าโอเอสที่ติดตั้งไว้สำหรับพัฒนาแอปพลิเคชันโดยเฉพาะจนเสร็จสมบูรณ์ จากนั้นจึงทำการรวมไฟล์ต่างๆของแอปพลิเคชันให้อยู่ในรูปของซิปไฟล์ (.Zip) แล้วนำไปส่งต่อให้กับผู้ดูแลระบบเพื่อนำไปใช้งานจริง
- 2) ผู้ดูแลระบบเข้าสู่ส่วนจัดการแอปพลิเคชันของโรซ่าโอเอสเพื่อทำการติดตั้งแอปพลิเคชัน
- 3) ผู้ดูแลระบบทำการอัปเดตแอปพลิเคชันเข้าสู่โรซ่าโอเอสผ่านส่วนจัดการแอปพลิเคชัน
- 4) ผู้ใช้ทุกคนจะสามารถใช้งานแอปพลิเคชันที่ผู้ดูแลระบบติดตั้งเข้าสู่ระบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การพัฒนาแอปพลิเคชันและการนำแอปพลิเคชันไปใช้งาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

4.1 ภาพรวมของระบบ

โรซ่าโอเอสเป็นเว็บแอปพลิเคชันซึ่งมีการทำงานพื้นฐานและลักษณะของส่วนติดต่อกับผู้ใช้ที่คล้ายคลึงกับระบบปฏิบัติการบนเดสก์ท็อป โดยโรซ่าโอเอสมีระบบจัดการข้อมูลของผู้ใช้ ระบบจัดการแอปพลิเคชันสำหรับผู้ดูแลระบบ และระบบสำหรับผู้ดูแลระบบ ซึ่งหลังจากที่ได้พัฒนาระบบเสร็จสิ้นแล้ว ระบบที่ได้นั้นสามารถนำไปใช้งานได้ทันทีพื้นฐาน และเป็นต้นแบบในการนำไปพัฒนาต่อยอดในภายภาคหน้าต่อไป

การพัฒนาโรซ่าโอเอสนี้มีจุดประสงค์หลักเพื่อนำไปติดตั้งบนเซิร์ฟเวอร์และใช้งานในเฉพาะกลุ่มหรือในเฉพาะองค์กร เพื่อให้เกิดความปลอดภัยทางด้านข้อมูลภายในองค์กร โดยผู้ดูแลระบบจะสามารถปรับแต่งค่าการใช้งานต่างๆของระบบให้เหมาะสมกับการใช้งานภายในองค์กรของตนได้ นอกจากนี้ระบบยังได้จัดเตรียม API ไว้เพื่อให้นักพัฒนาสามารถพัฒนาแอปพลิเคชันเพิ่มเติมขึ้นมาใช้งานและนำระบบไปพัฒนาต่อยอดได้อีกด้วย

ผู้ใช้งานโรซ่าโอเอสทุกคนจะมีบัญชีการใช้งานเป็นของตัวเองและได้รับพื้นที่การใช้งานส่วนตัว สามารถอัปโหลดไฟล์และใช้งานแอปพลิเคชันต่างๆได้ อีกทั้งเนื่องจากระบบมีการทำงานอยู่บนเว็บ ผู้ใช้จึงสามารถล็อกอินเข้าใช้ระบบได้จากทุกสถานที่ที่มีเว็บเบราว์เซอร์และอินเทอร์เน็ต

4.2 ความสามารถของระบบ

4.2.1 ความสามารถของระบบในส่วนของผู้ใช้ระบบ

ระบบในส่วนนี้ถูกออกแบบมาสำหรับการใช้งานทั่วไปที่ไม่มีความเกี่ยวข้องกับการจัดการดูแลระบบ ผู้ใช้งานระบบส่วนนี้อาจเป็นผู้ใช้ทั่วไปหรือผู้ดูแลระบบก็ได้

4.2.1.1 การสร้างบัญชีผู้ใช้งาน

ผู้ใช้ใหม่ที่ยังไม่มีบัญชีการใช้งานจำเป็นต้องสร้างบัญชีการใช้งานส่วนตัวขึ้นมา ก่อนเพื่อให้ได้รับพื้นที่สำหรับใช้งาน ดังรูปที่ 4.1

รูปที่ 4.1 การสร้างบัญชีผู้ใช้งาน

4.2.1.2 การตรวจสอบยืนยันตัวตนผู้ใช้

ผู้ใช้ที่มีบัญชีการใช้งานแล้วสามารถล็อกอินเข้าสู่ระบบเพื่อใช้งานระบบได้ โดยป้อน ชื่อผู้ใช้งาน (Username) และรหัสผ่าน (Password) ของผู้ใช้ เพื่อเข้าสู่ระบบ ดังรูปที่ 4.2

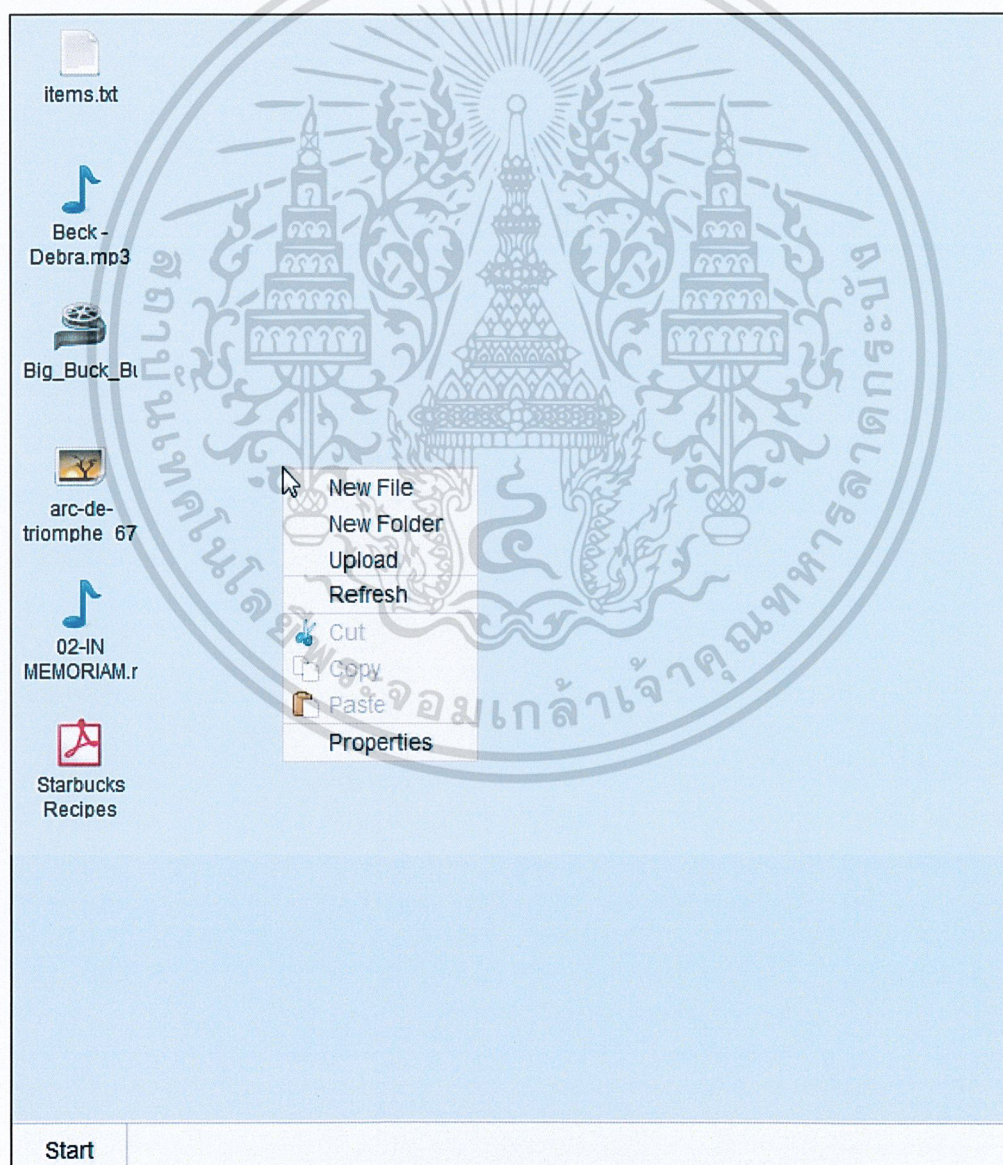
รูปที่ 4.2 หน้าจอการล็อกอินเข้าสู่ระบบของผู้ใช้ทั่วไป

เมื่อผู้ใช้ล็อกอินเข้าสู่ระบบแล้ว ระบบจะตรวจสอบว่าชื่อผู้ใช้งานและรหัสผ่านที่ผู้ใช้ระบุมีความถูกต้องหรือไม่ หากถูกต้องระบบจะอนุญาตให้ผู้ใช้เข้าใช้งานระบบได้ต่อไป แต่หากไม่ถูกต้องระบบจะปฏิเสธการเข้าใช้งาน และผู้ใช้จำเป็นจะต้องกรอกชื่อผู้ใช้งานและรหัสผ่านใหม่ให้ถูกต้อง

4.2.1.3 การจัดการไฟล์บนเดสก์ท็อปและเอ็กซ์พลอเรอร์

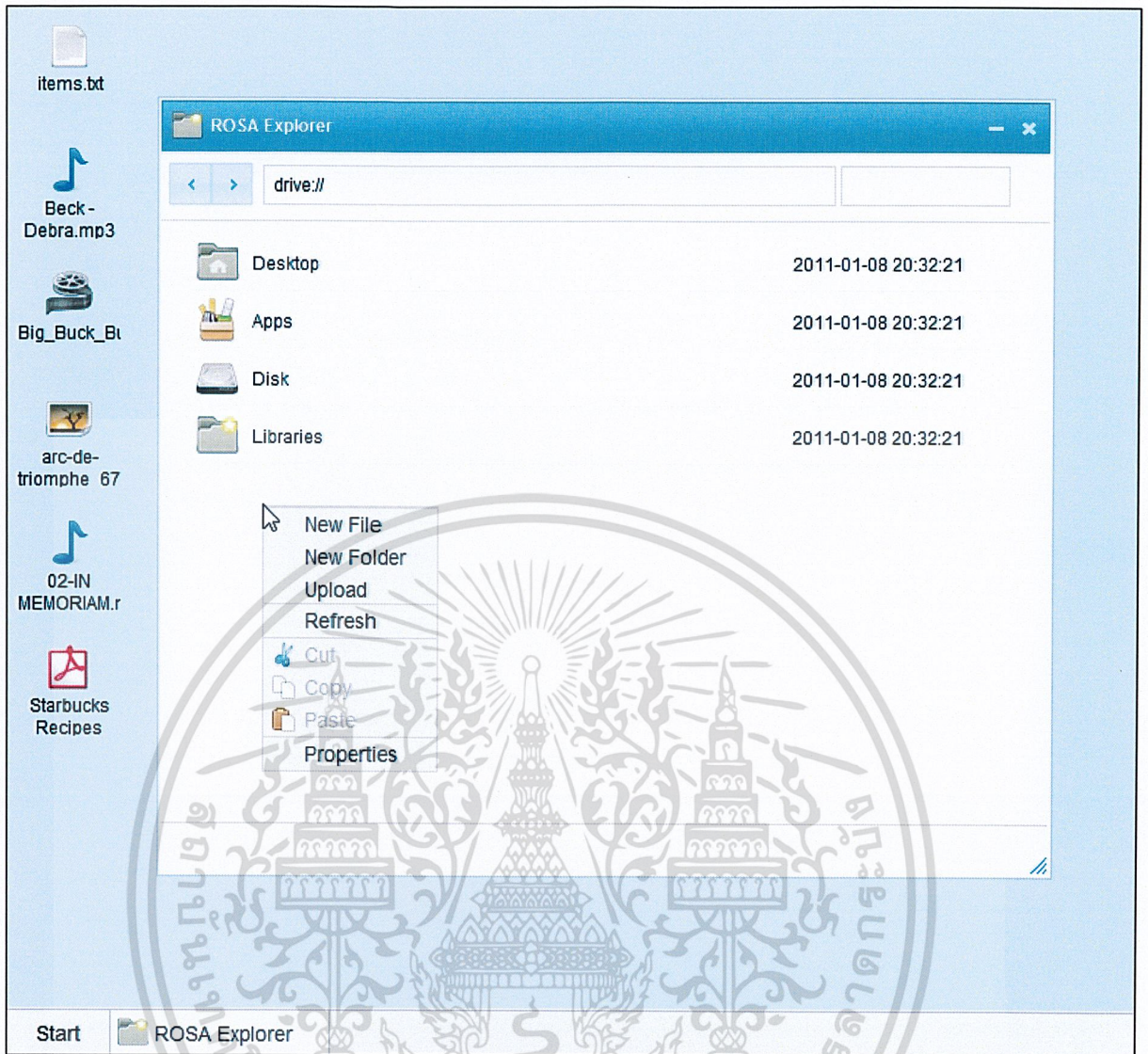
เดสก์ท็อปและเอ็กซ์พลอเรอร์ถือเป็นหนึ่งในแอปพลิเคชันพื้นฐานที่ระบบจัดเตรียมไว้ให้ และเป็นหน้าจอการใช้งานหลักของระบบ ซึ่งถูกออกแบบมาให้มีลักษณะคล้ายกับหน้าจอการใช้งานของระบบปฏิบัติการบนเดสก์ท็อป ผู้ใช้สามารถอัปโหลดหรือดาวน์โหลดไฟล์ สร้างหรือลบโฟลเดอร์ และจัดการไฟล์ต่างๆบนเดสก์ท็อปหรือในโฟลเดอร์ได้ โดยการจัดการไฟล์นั้นทำได้สะดวกเนื่องจากมีระบบ Drag และ Drop เพื่อย้ายที่อยู่ของไฟล์ รวมถึงระบบการเลือกไฟล์พร้อมกันหลายไฟล์โดยการลากเมาส์ครอบไฟล์อีกด้วย นอกจากนี้ผู้ใช้อังยังสามารถเรียกใช้งานแอปพลิเคชันต่างๆ ได้ด้วยการดับเบิลคลิกที่ไฟล์ที่ต้องการเพื่อเปิดการทำงานของแอปพลิเคชัน

ที่หน้าจอเดสก์ท็อปและเอ็กซ์พลอเรอร์ เมื่อผู้ใช้คลิกเมาส์ขวาที่พื้นที่ว่างบนหน้าจอ จะปรากฏเมนูการใช้งานซึ่งประกอบไปด้วยคำสั่งต่างๆ ดังรูปที่ 4.3 และ 4.4



รูปที่ 4.3 หน้าจอเดสก์ท็อป

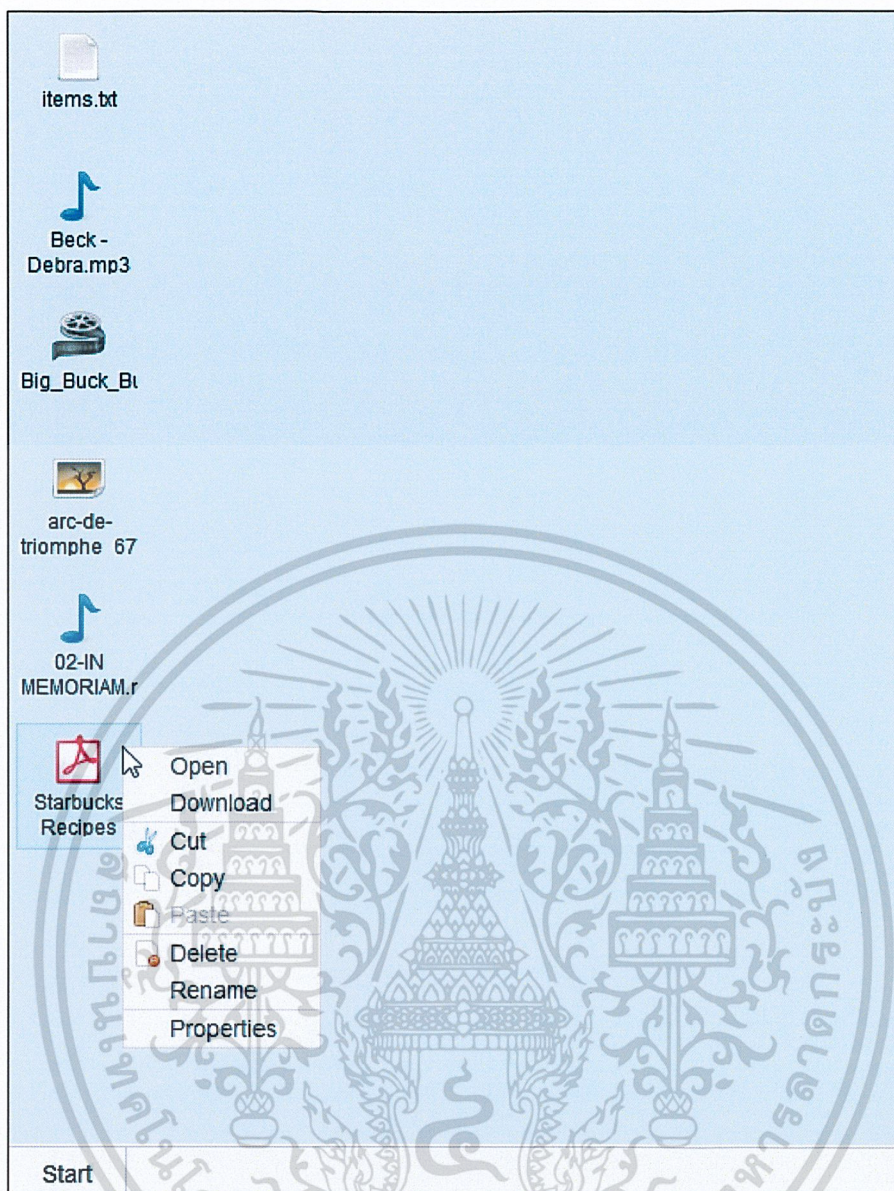
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 หน้าจอเอ็กซ์พลอเรอร์

คำสั่งต่างๆที่ปรากฏอยู่บนเมนูประกอบไปด้วยคำสั่ง New File, New Folder, Upload, Refresh, Cut, Copy, Paste และ Properties โดยสามารถศึกษารายละเอียดและการใช้งานแต่ละคำสั่งได้ในภาคผนวก ข.

นอกจากนี้เมื่อผู้ใช้คลิกเมาส์ขวาที่ไอคอนของไฟล์หรือโฟลเดอร์ จะปรากฏเมนูคำสั่ง ดังรูปที่ 4.5



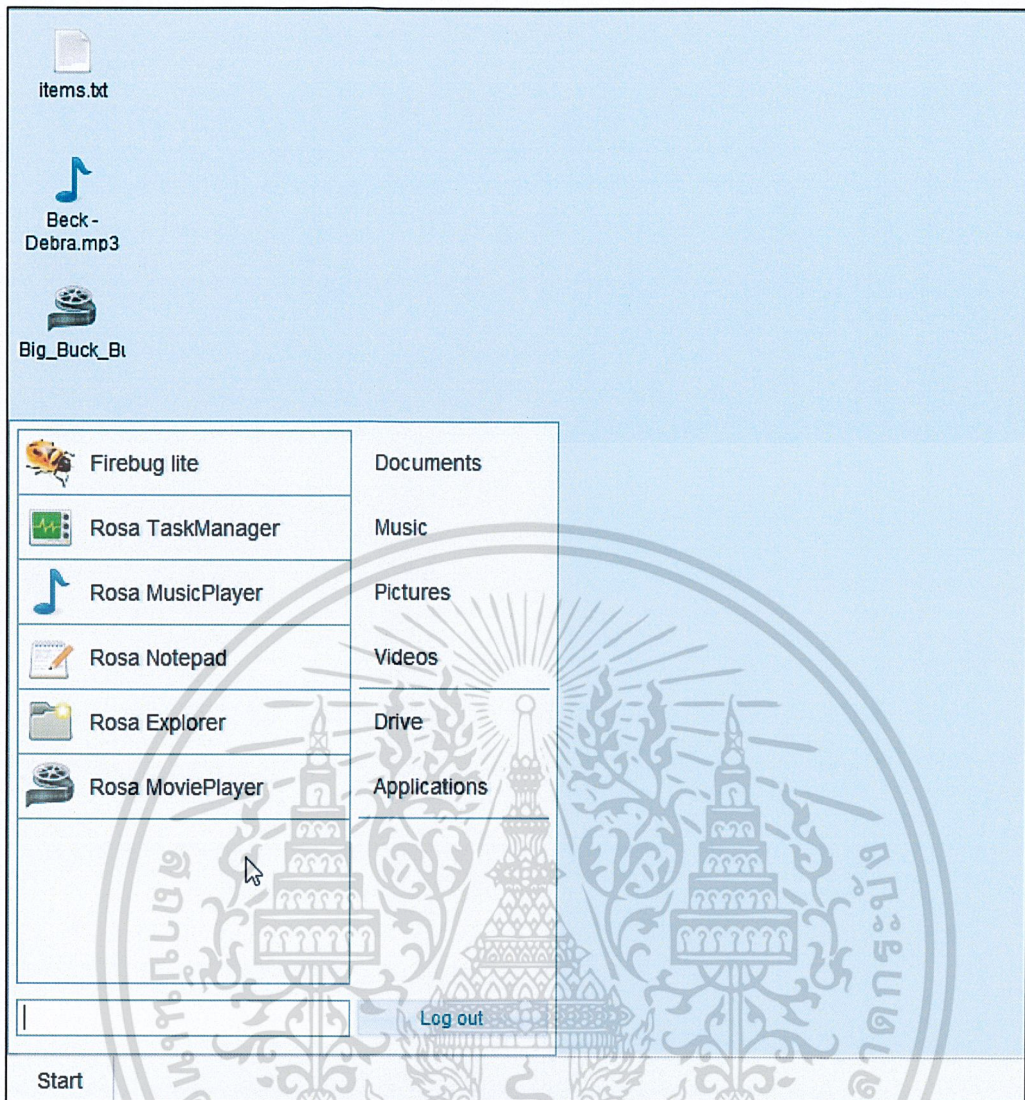
รูปที่ 4.5 เมนูคำสั่งที่ปรากฏขึ้นเมื่อกดคลิกเมาส์ขวาที่ไอคอนของไฟล์หรือโฟลเดอร์

คำสั่งต่างๆที่ปรากฏอยู่ในเมนูประกอบไปด้วยคำสั่ง Open, Download, Cut, Copy, Paste, Delete, Rename และ Properties โดยสามารถศึกษารายละเอียดและการใช้งานแต่ละคำสั่งได้ในภาคผนวก ข.

4.2.1.4 แอปพลิเคชันพื้นฐาน

ระบบได้จัดเตรียมแอปพลิเคชันสำหรับการใช้งานพื้นฐานไว้แล้ว เช่น โปรแกรม มัลติมีเดีย โปรแกรมประมวลผลคำ เป็นต้น ผู้ใช้สามารถเรียกใช้งานแอปพลิเคชันเหล่านี้ได้โดยกดปุ่ม Start ที่มุมด้านซ้ายล่างของหน้าจอเดสก์ทอป ดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แอปพลิเคชันพื้นฐานที่ระบบจัดเตรียมไว้ให้

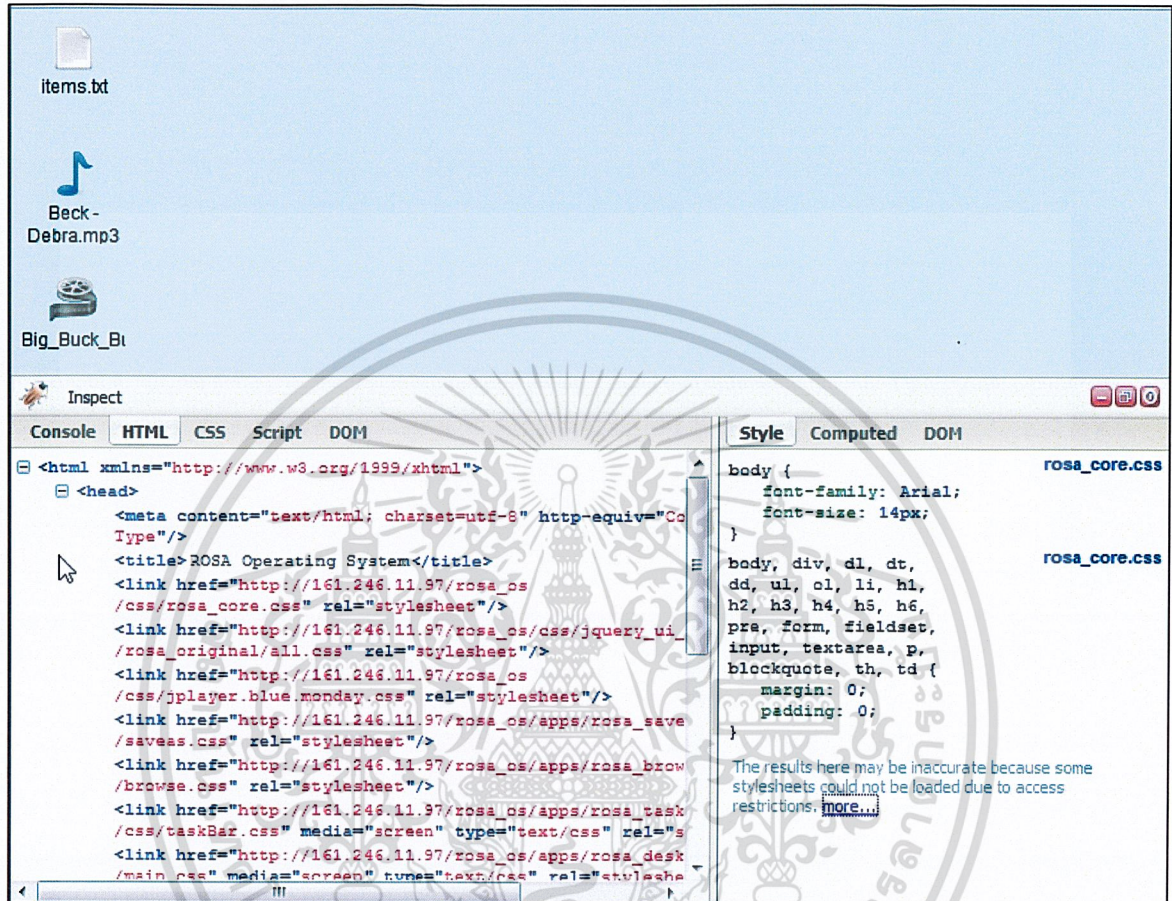
แอปพลิเคชันพื้นฐานทั้งหมดที่ระบบจัดเตรียมไว้ให้ประกอบด้วยดังนี้

Firebug Lite

พัฒนาโดย Joe Hewitt, johnjbarton, robcee, FirebugWorkingGroup และ Jan Odvarko โดย Firebug Lite เป็นแอปพลิเคชันที่อำนวยความสะดวกให้แก่ักพัฒนาที่สนใจการพัฒนาแอปพลิเคชันบนโรซ่าไอเอส การใช้งานนั้นนักพัฒนาสามารถเรียกดูและทดลองปรับแต่งโค้ดของเพจต่างๆได้ ซึ่งผลลัพธ์ของการปรับแต่งจะแสดงให้เห็นในทันที แต่จะไม่มีการบันทึกผลลัพธ์การปรับแต่งไว้แต่อย่างใด ดังนั้นจึงเหมาะสำหรับให้นักพัฒนาทำการทดลองปรับแต่งโค้ดเพื่อแก้ไขและทดสอบการทำงานของระบบ หรือทดสอบ API ของโรซ่าไอเอส นอกจากนี้แล้วนักพัฒนายังสามารถดูข้อมูลที่ส่งผ่านระหว่างไคลเอนต์และเซิร์ฟเวอร์ได้อีกด้วย จึงทำให้การวิเคราะห์และการแก้ปัญหาข้อผิดพลาดสามารถทำได้อย่างสะดวกและรวดเร็วมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอปพลิเคชัน Firebug Lite นี้เป็นตัวอย่างอีกหนึ่งความสามารถของระบบในการนำแอปพลิเคชันที่มีอยู่แล้วหรือแอปพลิเคชันที่พัฒนาขึ้นโดยผู้อื่นมาติดตั้งใช้งานภายในโรซ่าโอเอสได้ โดยตัวอย่างส่วนติดต่อของผู้ใช้ของ Firebug Lite นั้น สามารถแสดงให้เห็นได้ ดังรูปที่ 4.7

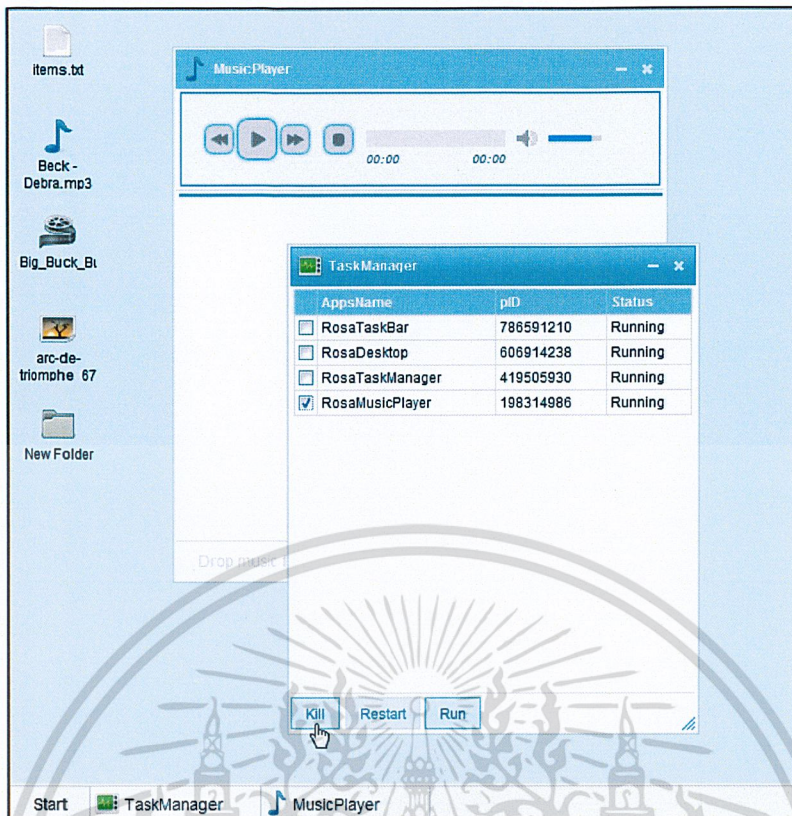


รูปที่ 4.7 Firebug Lite

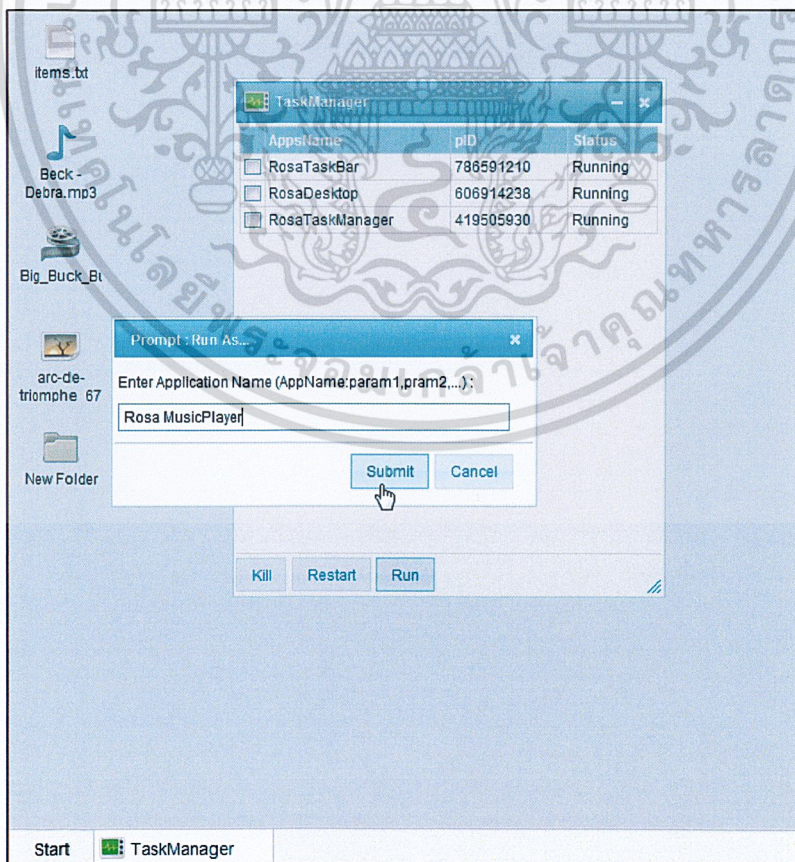
Rosa TaskManager

เป็นแอปพลิเคชันสำหรับติดตามดูรายละเอียดของแอปพลิเคชันที่กำลังทำงานอยู่ โดยรายละเอียดของแอปพลิเคชันที่แสดงในทาสก์เมเนเจอร์ประกอบด้วย ชื่อแอปพลิเคชัน (AppName), รหัสประจำตัวของโปรเซส (pID) และสถานะ (Status) ของแอปพลิเคชัน นอกจากนี้ ผู้ใช้ยังสามารถปิด, รีสตาร์ท หรือรันแอปพลิเคชันต่างๆ ได้ด้วยการกดปุ่ม Kill, Restart และ Run ตามลำดับ ดังรูปที่ 4.8 โดยหากผู้ใช้ต้องการรันแอปพลิเคชันใดๆ ผู้ใช้จำเป็นต้องป้อนชื่อของแอปพลิเคชันนั้น เช่นดังรูปที่ 4.9 เพื่อให้ระบบเริ่มต้นการทำงานของแอปพลิเคชันนั้นให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 Rosa TaskManager

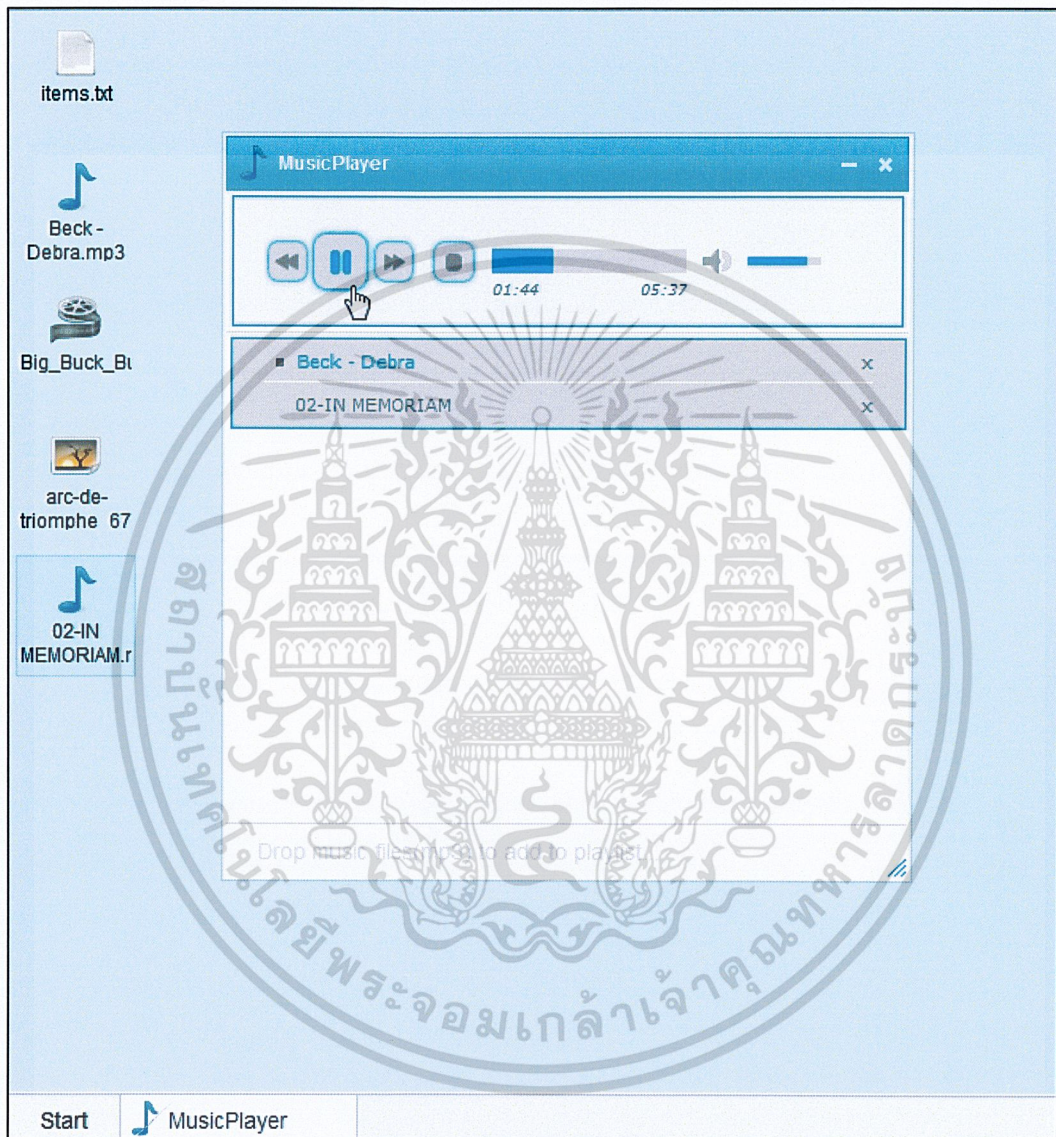


รูปที่ 4.9 การรันแอปพลิเคชันผ่านทาง Rosa TaskManager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Rosa MusicPlayer

เป็นแอปพลิเคชันสำหรับเล่นไฟล์เพลง ผู้ใช้สามารถเพิ่มและลบเพลงออกจาก Playlist, ปรับระดับเสียงเพลง และควบคุมการเล่นเพลงได้เช่นเดียวกับโปรแกรมเล่นไฟล์เพลงทั่วไป สามารถแสดงส่วนติดต่อกับผู้ใช้ของ Rosa MusicPlayer ได้ ดังรูปที่ 4.10

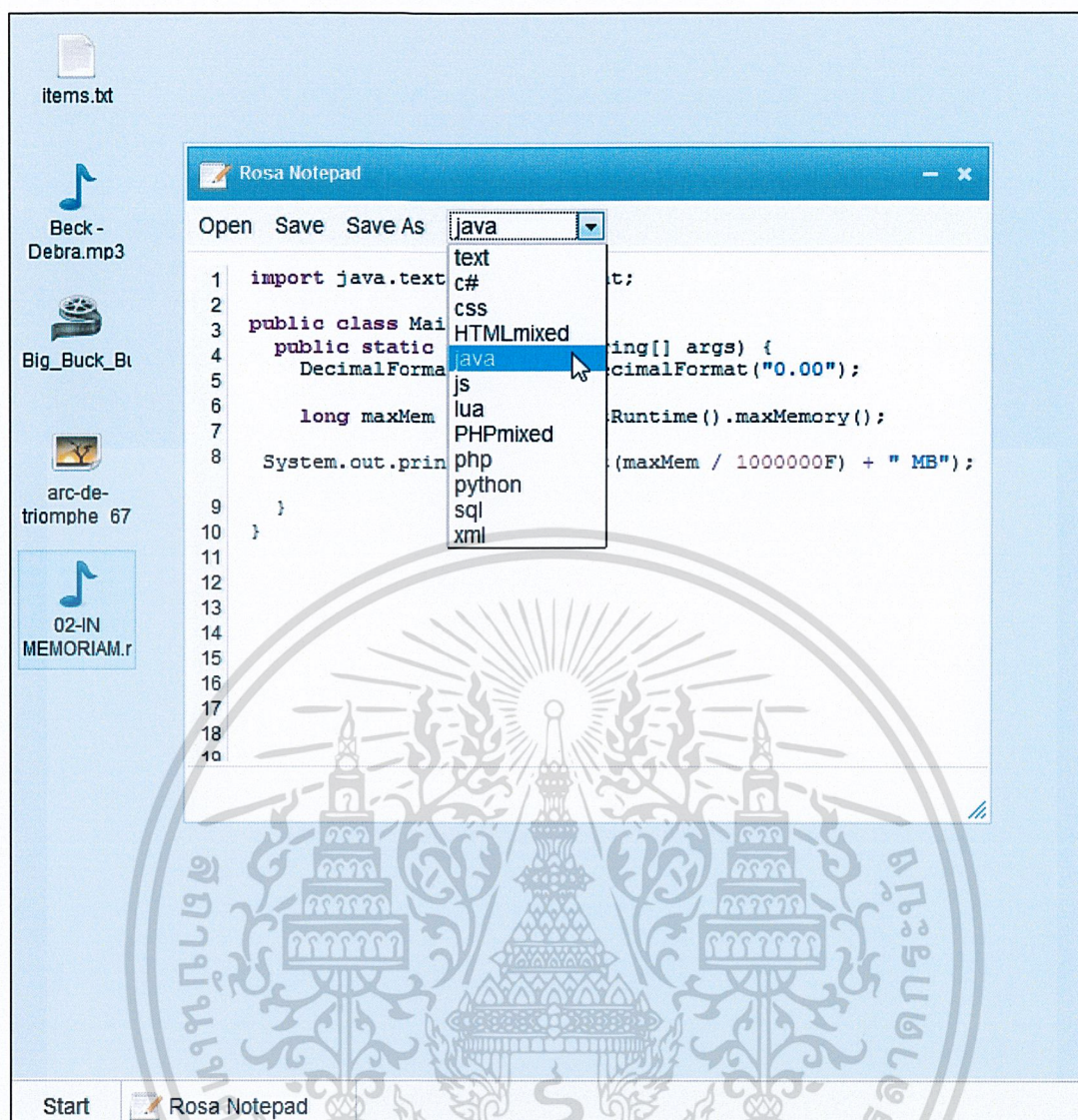


รูปที่ 4.10 Rosa MusicPlayer

Rosa Notepad

เป็นแอปพลิเคชันในการประมวลผลคำ โดยผู้ใช้สามารถเลือกประเภทของการแสดงผลให้เหมาะสมกับข้อความได้ เช่น ข้อความธรรมดา, โค้ดโปรแกรมภาษา Java, C#, HTML, PHP หรือ SQL เป็นต้น โดยสามารถแสดงส่วนติดต่อกับผู้ใช้ของ Rosa Notepad ได้ ดังรูปที่ 4.11

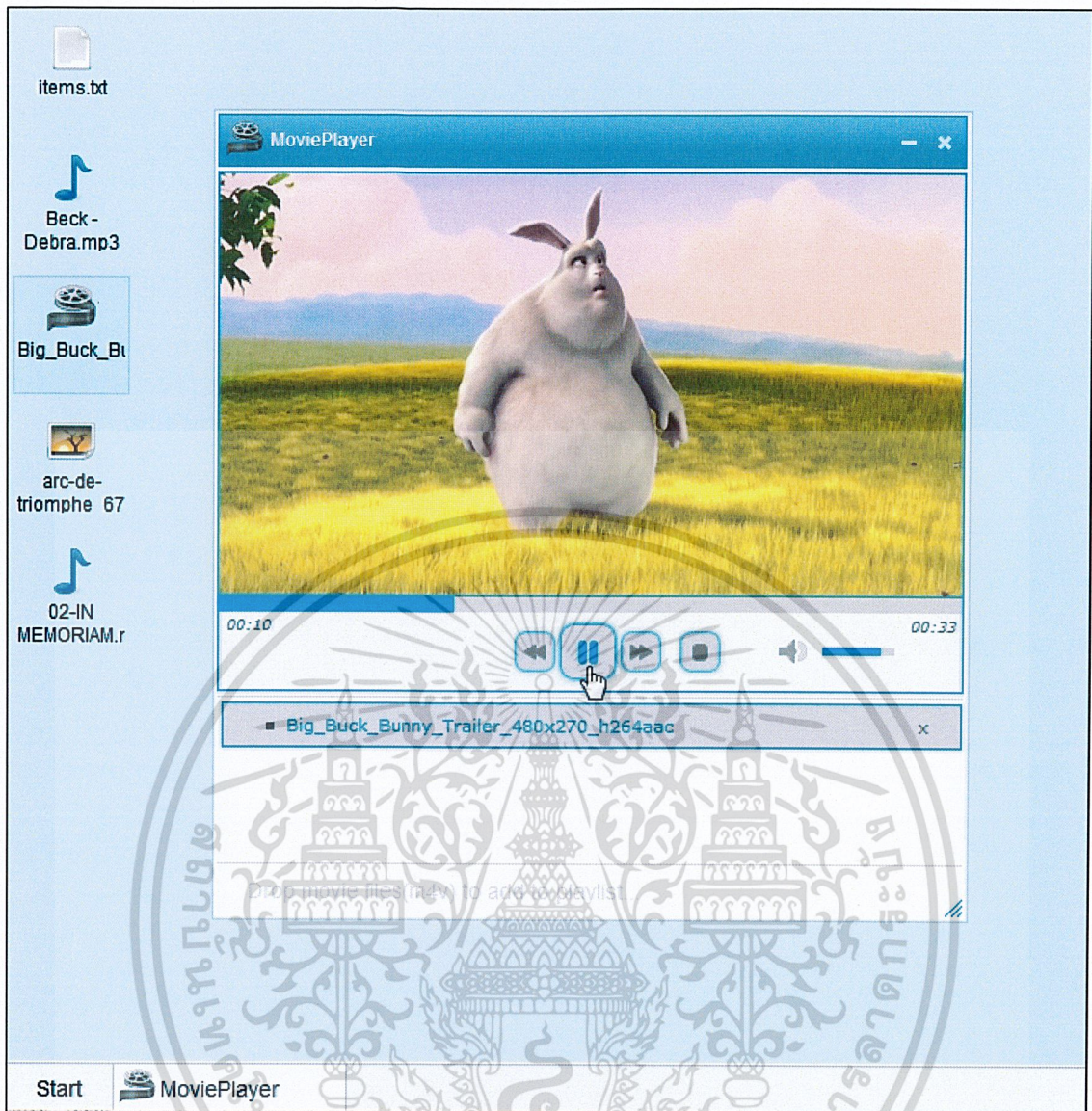
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 Rosa Notepad

Rosa MoviePlayer

เป็นแอปพลิเคชันในการเล่นไฟล์วิดีโอ โดยผู้ใช้สามารถเพิ่มและลบไฟล์วิดีโอออกจาก Playlist, ปรับระดับเสียงวิดีโอ และควบคุมการเล่นวิดีโอได้เช่นเดียวกัน โปรแกรมเล่นไฟล์วิดีโอทั่วไป โดยสามารถแสดงส่วนติดต่อกับผู้ใช้ของ Rosa MoviePlayer ได้ ดังรูปที่ 4.12

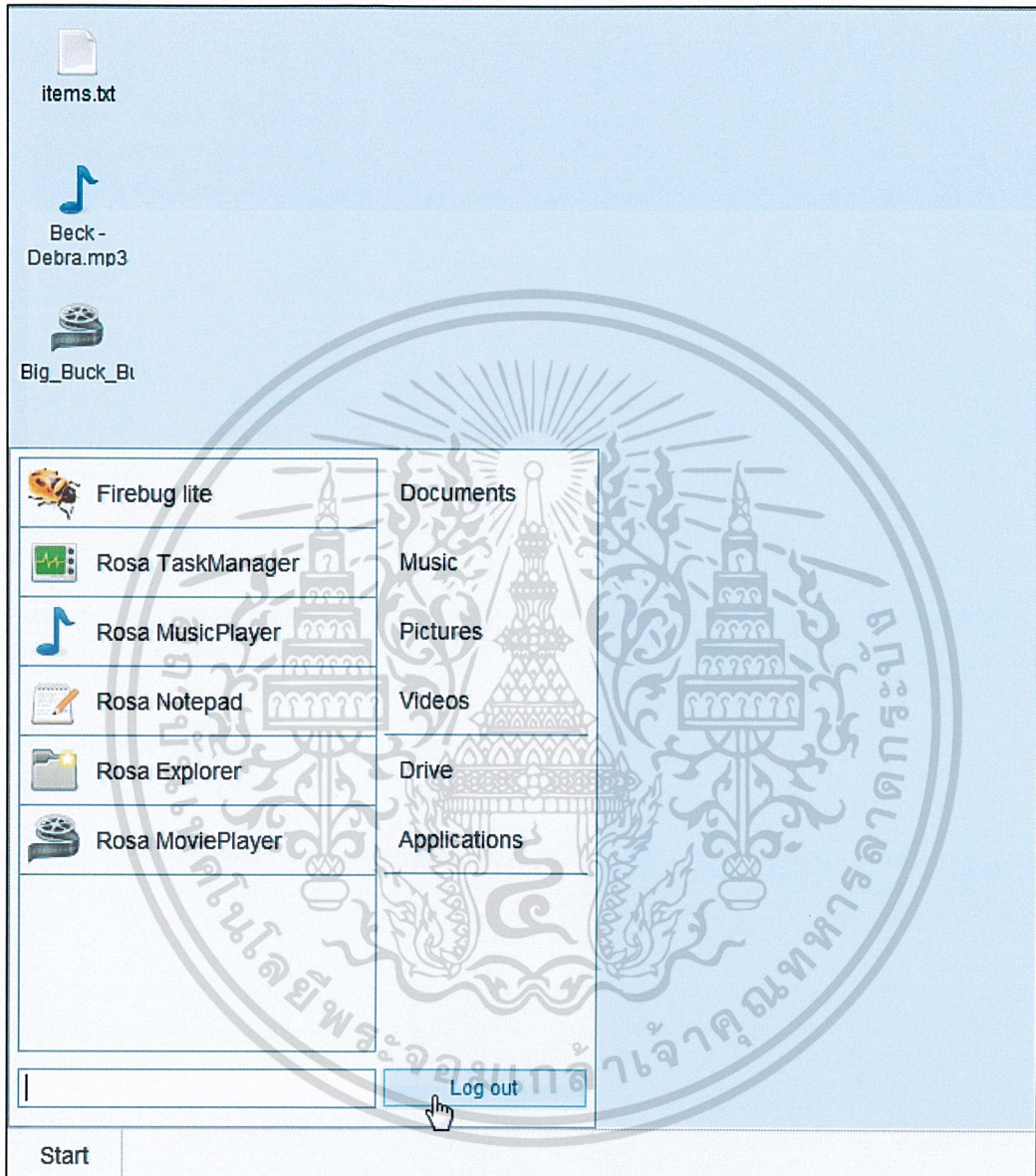


รูปที่ 4.12 Rosa MoviePlayer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.5 การลือกเอาที่ออกจากระบบ

เมื่อสิ้นสุดการใช้งาน ผู้ใช้สามารถออกจากระบบได้โดยการคลิกปุ่มลือกเอาที่ออกจากระบบ ดังรูปที่ 4.13



รูปที่ 4.13 การลือกเอาที่ออกจากระบบ

4.2.2 ความสามารถของระบบในส่วนของผู้ดูแลระบบ

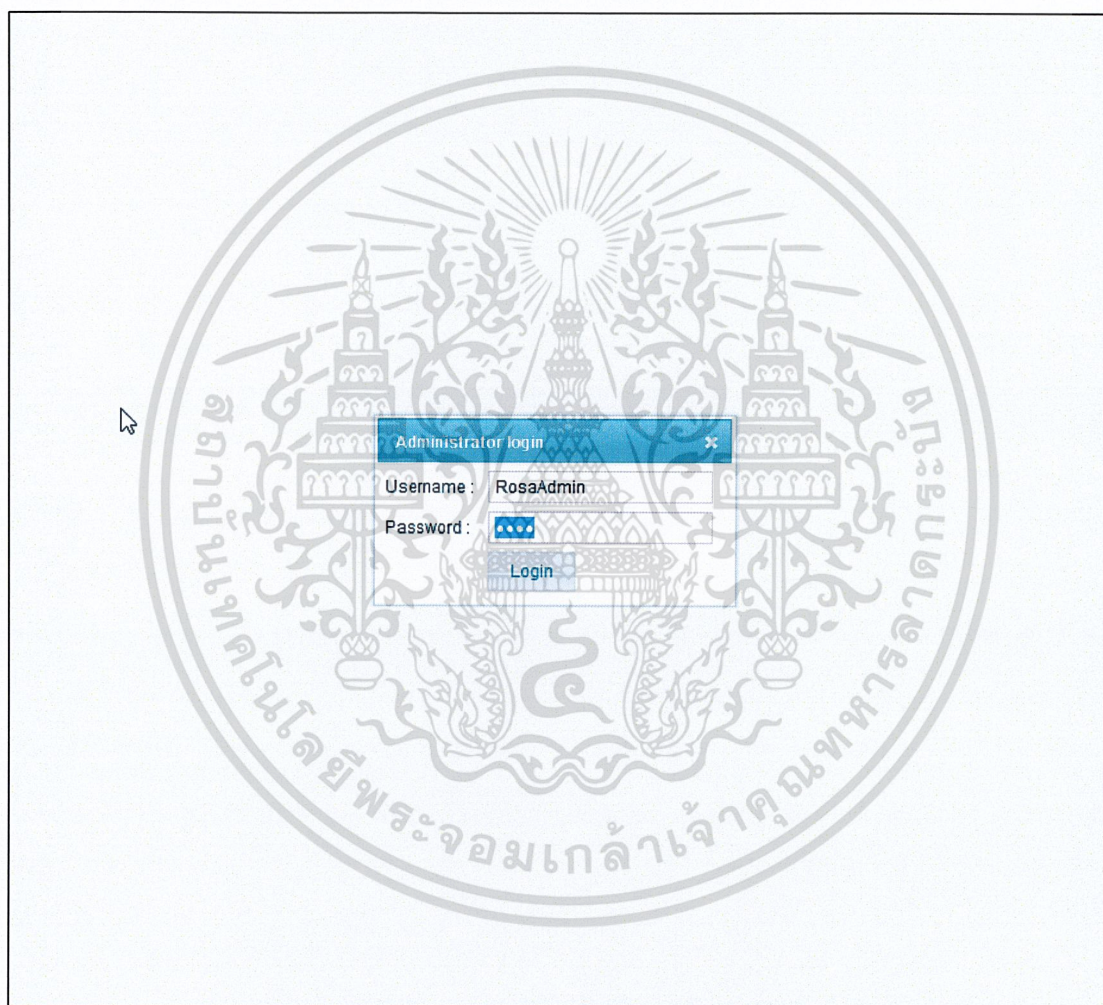
ระบบในส่วนนี้ถูกออกแบบมาสำหรับผู้ดูแลระบบ นั่นคือผู้ใช้ที่เป็นสมาชิกของกลุ่มผู้ใช้ “Administrator” โดยเป็นส่วนที่มีไว้สำหรับใช้ในการจัดการดูแลข้อมูลต่างๆของระบบ ซึ่งแบ่งออกเป็น ส่วนของข้อมูลที่เกี่ยวข้องกับผู้ใช้งาน และส่วนของข้อมูลที่เกี่ยวข้องกับแอปพลิเคชัน สมาชิกในกลุ่มผู้ดูแลระบบจะได้รับพื้นที่การใช้งานส่วนตัวเช่นเดียวกับผู้ใช้งานทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.1 การตรวจสอบสิทธิของผู้ดูแลระบบ

การล็อกอินเข้าสู่ส่วนการใช้งานของผู้ดูแลระบบนั้นจะกระทำเช่นเดียวกับการล็อกอินเข้าสู่ระบบของผู้ใช้งานทั่วไป โดยหากระบบตรวจสอบแล้วว่าผู้ใช้เป็นสมาชิกของกลุ่ม “Administrator” หรือกลุ่มของผู้ดูแลระบบ ผู้ใช้ก็จะมีสิทธิเข้าสู่ส่วนของผู้ดูแลระบบต่อไป

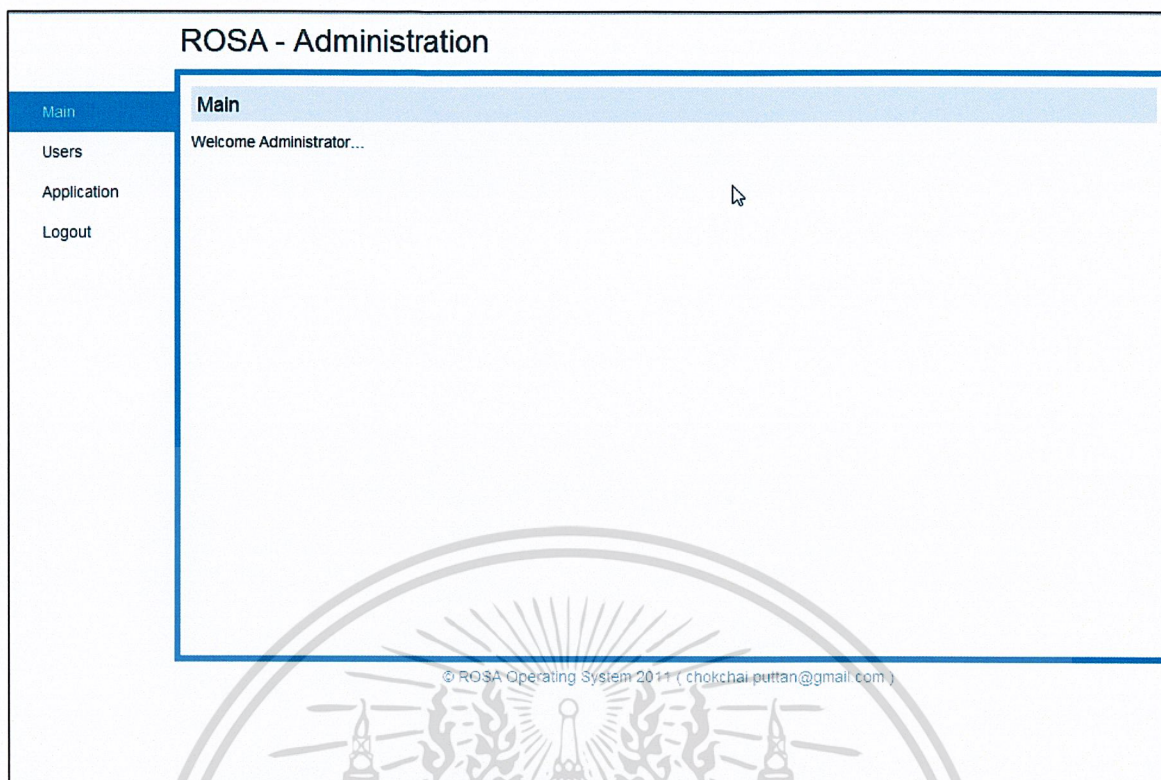
หน้าจอการล็อกอินเข้าสู่ส่วนการใช้งานของผู้ดูแลระบบ สามารถแสดงได้ดังรูปที่ 4.14 โดยผู้ใช้จำเป็นต้องกรอกชื่อผู้ใช้งานและรหัสผ่านของบัญชีผู้ใช้งานที่มีสิทธิของผู้ดูแลระบบให้ถูกต้อง



รูปที่ 4.14 หน้าจอการล็อกอินเข้าสู่ระบบของผู้ดูแลระบบ

เมื่อระบบตรวจสอบแล้วว่าบัญชีผู้ใช้งานนั้นเป็นสมาชิกของกลุ่มผู้ดูแลระบบ ระบบก็จะนำผู้ใช้เข้าสู่ส่วนหน้าจอการใช้งานหลักของผู้ดูแลระบบ ซึ่งสามารถแสดงได้ดังรูปที่ 4.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 หน้าจอการใช้งานของส่วนของผู้ดูแลระบบ

4.2.2.2 การจัดการข้อมูลสำหรับผู้ดูแลระบบ

ระบบในส่วนนี้สามารถแบ่งออกได้เป็น 2 ส่วนคือ ส่วนของข้อมูลที่เกี่ยวข้องกับ
ผู้ใช้งาน และส่วนของข้อมูลที่เกี่ยวข้องกับแอปพลิเคชัน

1) ส่วนของข้อมูลที่เกี่ยวข้องกับผู้ใช้

ส่วนนี้จะแสดงข้อมูลต่างๆที่เกี่ยวข้องกับผู้ใช้ ดังรูปที่ 4.16 โดยผู้ดูแลระบบ
สามารถแก้ไข ลบ และสืบค้นข้อมูลที่เกี่ยวข้องกับผู้ใช้ได้ในส่วนนี้

ROSA - Administration										
Main	Users									
Users									<input type="text"/>	Search
Application	#	Username	Group	Email	Used Quota	Remain Quota	Max Quota	Edit	Delete	
Logout	1	chokchai	administrator	chokchai.puttan@gmail.com	4 MB	1020 MB	1024 MB	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>	
	2	RosaAdmin	administrator	clarinet_chan@hotmail.com	12 MB	1012 MB	1024 MB	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>	
	7	Rosetta	user	clarinet_chan@hotmail.com	0 MB	1024 MB	1024 MB	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>	

© ROSA Operating System 2011 (chokchai.puttan@gmail.com)

รูปที่ 4.16 ส่วนของข้อมูลที่เกี่ยวข้องกับผู้ใช้ในหน้าจอการใช้งานของผู้ดูแลระบบ

ข้อมูลเกี่ยวกับผู้ใช้ที่แสดงในส่วนนี้ประกอบไปด้วย

- # (ID): รหัสประจำตัวผู้ใช้
- Username: ชื่อที่ผู้ใช้ได้ลงทะเบียนไว้
- Group: กลุ่มที่ผู้ใช้เป็นสมาชิกอยู่
- Email: อีเมลแอดเดรสของผู้ใช้
- Used Quota: ขนาดพื้นที่การใช้งานที่ผู้ใช้ได้ใช้งานไปแล้ว
- Remain Quota: ขนาดพื้นที่การใช้งานที่เหลือของผู้ใช้
- Max Quota: ขนาดพื้นที่การใช้งานสูงสุดของผู้ใช้

สามารถศึกษารายละเอียดคำอธิบายของข้อมูลเหล่านี้เพิ่มเติมได้ในภาคผนวก ก.

หากผู้ดูแลระบบต้องการแก้ไขข้อมูลที่เกี่ยวข้องกับผู้ใช้ ระบบจะแสดงหน้าจอการใช้งาน ดังรูปที่ 4.17

ROSA - Administration

Main

Users

Application

Logout

Users > Edit User

ID :

Username :

Email :

Quota : MB

Group :

© ROSA Operating System 2011 (chokchai.puttan@gmail.com)

รูปที่ 4.17 การแก้ไขข้อมูลที่เกี่ยวข้องกับผู้ใช้ในหน้าจอการใช้งานของผู้ดูแลระบบ

ข้อมูลผู้ใช้ที่ผู้ดูแลระบบสามารถแก้ไขได้ ประกอบด้วยดังนี้

- Quota: ขนาดพื้นที่การใช้งานสูงสุดของผู้ใช้
- Group: กลุ่มที่ผู้ใช้เป็นสมาชิกอยู่

สามารถศึกษาวิธีการแก้ไขข้อมูลและรายละเอียดคำอธิบายของข้อมูลเหล่านี้ได้ใน

ภาคผนวก ก.

2) ส่วนของข้อมูลที่เกี่ยวข้องกับแอปพลิเคชัน

ในส่วนนี้จะแสดงข้อมูลต่างๆที่เกี่ยวข้องกับแอปพลิเคชัน ดังรูปที่ 4.18 โดยผู้ดูแล

ระบบสามารถอัปเดตแอปพลิเคชัน เปลี่ยนแปลงสถานะของแอปพลิเคชัน ลบแอปพลิเคชัน และสืบค้นข้อมูลที่เกี่ยวข้องกับแอปพลิเคชันได้ในส่วนนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROSA - Administration

Main

Users

Application

Logout

Application

Name	Version	Developer	Status	Delete
Rosa Taskbar			Enable ▾	<input type="button" value="Delete"/>
FireBug Lite			Enable ▾	<input type="button" value="Delete"/>
rosa_ui			Enable ▾	<input type="button" value="Delete"/>
rosa_taskmanager			Enable ▾	<input type="button" value="Delete"/>
rosa_musicplayer			Enable ▾	<input type="button" value="Delete"/>
Rosa Notepad	1.0	Chokchai Puttan	Enable ▾	<input type="button" value="Delete"/>
rosa_desktop			Enable ▾	<input type="button" value="Delete"/>
rosa_saveas			Enable ▾	<input type="button" value="Delete"/>
rosa_viewer			Enable ▾	<input type="button" value="Delete"/>
RosaExplorer			Enable ▾	<input type="button" value="Delete"/>
rosa_movieplayer			Enable ▾	<input type="button" value="Delete"/>
rosa_upload			Enable ▾	<input type="button" value="Delete"/>
rosa_browse			Enable ▾	<input type="button" value="Delete"/>
rosa_rename			Enable ▾	<input type="button" value="Delete"/>

© ROSA Operating System 2011 (chokchai.puttan@gmail.com)

รูปที่ 4.18 ส่วนของข้อมูลที่เกี่ยวข้องกับแอปพลิเคชันในหน้าจอการใช้งานของผู้ดูแลระบบ

ข้อมูลเกี่ยวกับแอปพลิเคชันที่แสดงในส่วนนี้ประกอบไปด้วย

- Name: ชื่อแอปพลิเคชัน
- Version: เวอร์ชันของแอปพลิเคชัน
- Developer: ชื่อผู้พัฒนาแอปพลิเคชัน
- Status: สถานะของแอปพลิเคชัน

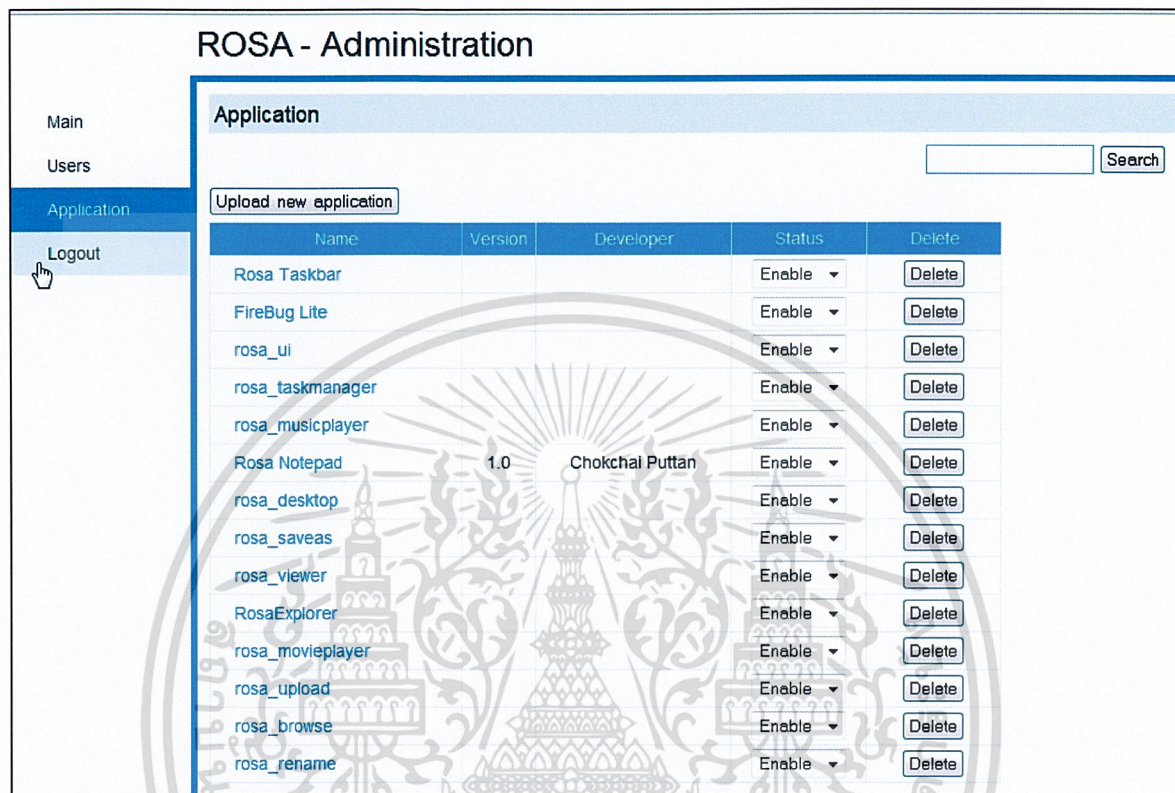
โดยสามารถศึกษาการใช้งานและรายละเอียดคำอธิบายของข้อมูลเหล่านี้เพิ่มเติมได้

ในภาคผนวก ก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.3 การลือกเอาที่ออกจากระบบ

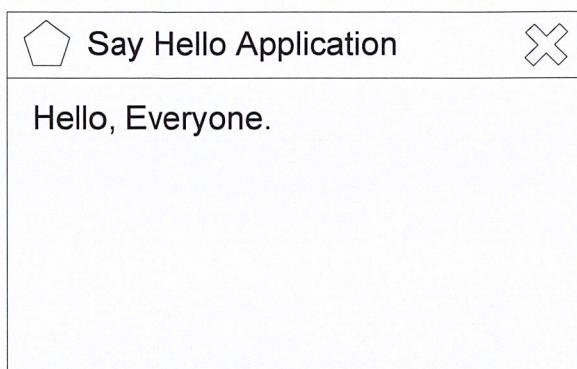
เมื่อสิ้นสุดการใช้งาน ผู้ดูแลระบบสามารถออกจากระบบได้โดยการลือกเอาที่ออกจากระบบ ดังรูปที่ 4.19



รูปที่ 4.19 การลือกเอาที่ออกจากส่วนของผู้ดูแลระบบ

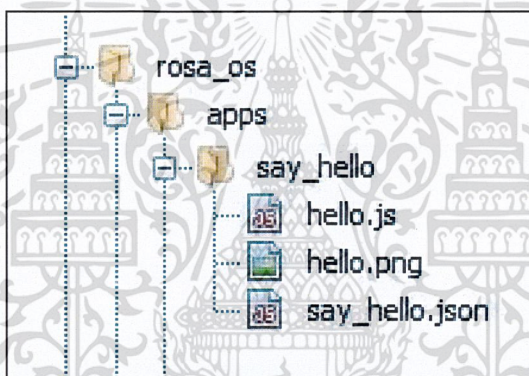
4.2.1 การพัฒนาแอปพลิเคชันสำหรับโรซ่าโอเอส

ในส่วนนี้จะป็นตัวอย่างการพัฒนาแอปพลิเคชันรูปแบบพื้นฐานเพื่อแสดงให้เห็กระบวนการในการสร้างแอปพลิเคชันว่าสามารถทำได้ยัอย่างไร โดยแอปพลิเคชันที่จะสร้างนี้มีชื่อว่ “SayHello” ซึ่งมีส่วนติดต่อกับผู้ใชัดังรูปที่ 4.20 ตัวอย่างการพัฒนานี้เป็นแบบย่อ หากต้องการรายละเอียดเพิ่มเติมเกี่ยวกับการพัฒนาแอปพลิเคชันบนโรซ่าโอเอส สามารถศึกษารายละเอียดได้จากภาคผนวก จ.



รูปที่ 4.20 ต้นแบบส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน SayHello

ในการพัฒนาแอปพลิเคชันบนโรซ่าโอเอสนั้น ก่อนอื่นจะต้องสร้างโฟลเดอร์สำหรับเก็บไฟล์ของแอปพลิเคชันไว้ในโฟลเดอร์ “rosa_os/apps/” โดยในที่นี้จะทำการสร้างโฟลเดอร์ที่ชื่อ “say_hello” ซึ่งและมีไฟล์ต่างๆสำหรับทำงานดังรูปที่ 4.21



รูปที่ 4.21 โครงสร้างไฟล์ของแอปพลิเคชัน Say Hello

- **hello.js:** เป็นไฟล์จาวาสคริปต์ที่ใช้สำหรับการเขียนโปรแกรมการทำงานของแอปพลิเคชัน
- **hello.png:** เป็นไฟล์รูปภาพสำหรับใช้เป็นไอคอนของแอปพลิเคชัน
- **say_hello.json:** เป็นไฟล์สำหรับอธิบายแอปพลิเคชันให้กับระบบ เพื่อนำแอปพลิเคชันนี้ไปแสดงให้ผู้ใช้งานได้เห็นและสามารถใช้งานได้

จะเห็นว่าไฟล์สำหรับอธิบายแอปพลิเคชัน หรือ say_hello.json นั้นจะมีชื่อเหมือนกับชื่อโฟลเดอร์หลักของแอปพลิเคชัน ทั้งนี้เพื่อให้โรซ่าโอเอสสามารถทราบได้ว่าเป็นไฟล์สำหรับอธิบายแอปพลิเคชัน (Application Description) โดยซอร์สโค้ดของไฟล์นั้นเป็นดังรูปที่ 4.22

```

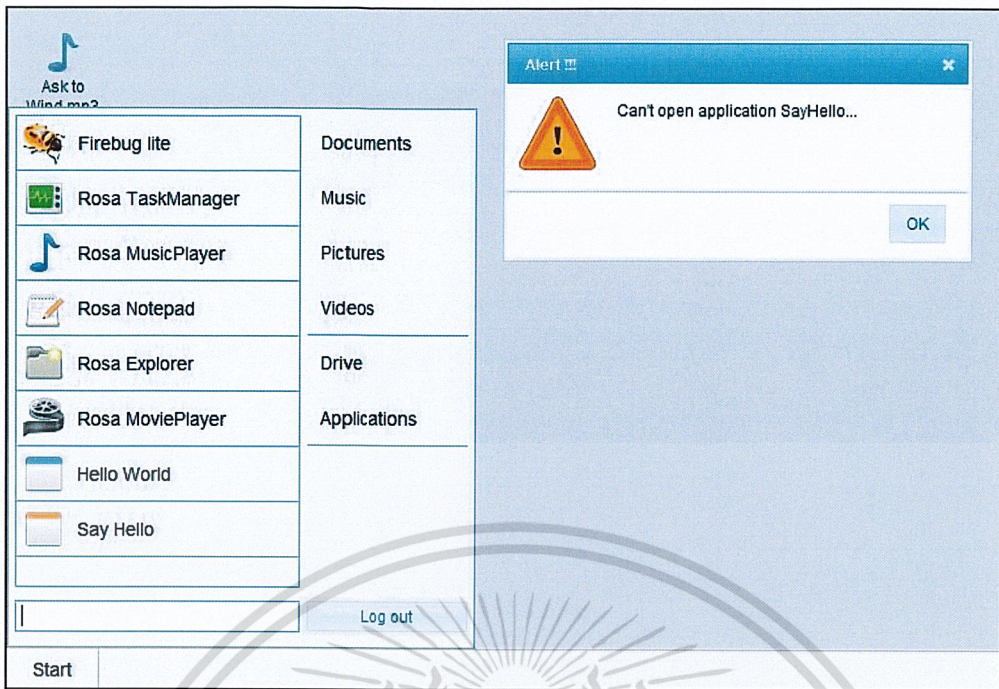
{
  "icon" : {
    "title" : "Say Hello",
    "iconImage" : "hello.png"
  },
  "onStart" : {
    "appName" : "SayHello",
    "param" : []
  },
  "js" : [ "hello" ]
}

```

รูปที่ 4.22 ซอร์สโค้ดของไฟล์ say_hello.json

- **icon:** ไอคอนที่ใช้แสดงในส่วนสตาร์ทเมนู โดยจะมีชื่อ (title) คือ “Say Hello” และจะใช้ไฟล์ “hello.png” เป็นรูปไอคอน (iconImage)
- **onStart:** เมื่อผู้ใช้คลิกเพื่อให้แอปพลิเคชันเริ่มการทำงาน ระบบเรียกจะแอปพลิเคชันที่ชื่อ “SayHello” โดยในที่นี้ไม่มีการส่งพารามิเตอร์สำหรับเริ่มต้น (param)
- **js:** หมายถึง ให้มีการนำไฟล์จาวาสคริปต์ชื่อ “hello” (hello.js) เข้ามาทำงานในระบบเมื่อโรซ่าโอเอสเริ่มทำงาน

หลังจากที่เขียนไฟล์สำหรับอธิบายแอปพลิเคชันแล้ว จะทดสอบแอปพลิเคชันโดยการเข้าใช้งานโรซ่าโอเอส ซึ่งจะพบว่าแอปพลิเคชันนั้นถูกนำมาแสดงที่สตาร์ทเมนูแล้ว โดยมีชื่อว่า Say Hello และมีการแสดงรูปตามที่ได้กำหนดไว้ แต่แอปพลิเคชันยังไม่สามารถทำงานได้ เนื่องจากยังไม่มีเขียนแอปพลิเคชันที่ชื่อว่า “SayHello” ระบบจึงแสดงข้อความแจ้งเตือนว่า ไม่สามารถเรียกใช้งานแอปพลิเคชัน “SayHello” ได้ (Can’t open application SayHello...) ดังรูปที่ 4.23



รูปที่ 4.23 ผลที่ได้จากการเขียนไฟล์สำหรับอธิบายแอปพลิเคชันของแอปพลิเคชัน SayHello

จากนั้นแก้ไขไฟล์จาวาสคริปต์ “hello.js” โดยสร้างแอปพลิเคชันออบเจกต์ชื่อ SayHello ต่อมากำหนดชื่อของแอปพลิเคชัน (NAME) และฟังก์ชันทำงานของแอปพลิเคชัน (run) จากนั้นให้นำแอปพลิเคชันออบเจกต์มาลงทะเบียนกับส่วนจัดการแอปพลิเคชัน เพื่อให้แอปพลิเคชันสามารถเริ่มทำงานได้ (คำสั่ง R.ApplicationManager.add(SayHello)) ดังรูปที่ 4.24

```
(function($, window, undefined) {
    var R = window.R;

    //Application Object
    var SayHello = {
        NAME: 'SayHello', //Application name

        run: function() {
            // ...
        }
    };

    //add Application Object to Application Manager
    R.ApplicationManager.add(SayHello);
})(jQuery, window);
```

รูปที่ 4.24 ซอร์สโค้ดของแอปพลิเคชันออบเจกต์ที่ได้ลงทะเบียนกับส่วนจัดการแอปพลิเคชันแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะเริ่มเขียนฟังก์ชันการทำงานให้กับแอปพลิเคชัน โดยจะเขียนขึ้นในฟังก์ชันทำงาน (run) ซึ่งจะเพิ่ม HTML ที่มีข้อความ “Hello, Everyone.” ลงบนเดสก์ท็อปด้วยโรซ่า API หลังจากที่ได้เพิ่มแล้วระบบคืนค่าออบเจกต์ที่เรียกว่า ADA (Application DOM Area) ซึ่งเป็นออบเจกต์ของ jQuery ที่ใช้สำหรับกำหนดขอบเขตของ DOM ในส่วนของแอปพลิเคชัน เปลี่ยน ADA ที่ได้มาให้กลายเป็นหน้าต่างของแอปพลิเคชัน โดยในขั้นสุดท้ายนั้นจำเป็นต้องกำหนดค่าของ ADA โดยใช้ชื่อว่า \$R ให้เป็นสมาชิกในแอปพลิเคชันออบเจกต์ด้วย เพื่อให้แอปพลิเคชันสามารถทำงานร่วมกับทาสก์บาร์ได้ ดังรูปที่ 4.25

```
(function($, window, undefined){

    var R = window.R;

    //Application Object
    var SayHello = {

        NAME: 'SayHello', //Application name

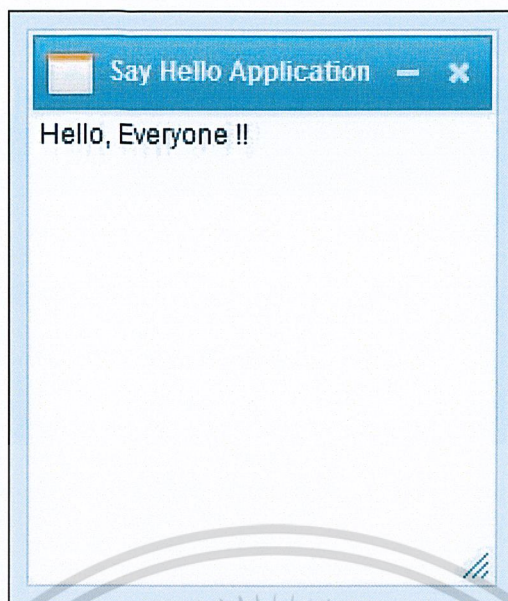
        run: function(){
            //add HTML to RosaDesktop and create ADA (Application DOM Area) from HTML
            var $R = R.addHTML('<div>Hello, Everyone !!</div>');
            //create RosaDialog
            this.$R = $R.ui().rosaDialog({
                iconClass: 'rosa_icon_app_default_orange_24', //set dialog icon
                title: 'Say Hello Application', //set dialog title
                width: 200, //set height (px)
                height: 200 //set width (px)
            });
        }
    };

    //add Application Object to Application Manager
    R.ApplicationManager.add(SayHello);

})(jQuery, window);
```

รูปที่ 4.25 ซอร์สโค้ดเพิ่มเติมฟังก์ชันการทำงานที่อยู่ในแอปพลิเคชันออบเจกต์

จากนั้นทดสอบแอปพลิเคชันบนโรซ่าโอเอส หากแอปพลิเคชันสามารถใช้งานได้ แอปพลิเคชันนั้นจะแสดงผลตามที่ได้ออกแบบไว้ตั้งแต่ตอนต้น และยังสามารถทำงานร่วมกับทาสก์บาร์ได้อย่างไม่มีปัญหาอีกด้วย ดังรูปที่ 4.26



รูปที่ 4.26 ส่วนติดต่อกับผู้ใช้ของแอปพลิเคชัน Say Hello

หลังจากที่พัฒนาแอปพลิเคชันและทดสอบจนมั่นใจว่าไม่มีข้อผิดพลาดใดๆแล้ว เราก็สามารถนำแอปพลิเคชันที่พัฒนาขึ้นมาไปรวมไฟล์ (Pack) ให้อยู่ในรูปแบบของซิปไฟล์ (Zip) เพื่อที่จะได้นำไปแจกจ่ายให้ผู้อื่นนำไปติดตั้งบนระบบ โรซ่า ไอเอส โดยการซิปไฟล์นั้นจะทำการซิปจากโฟลเดอร์ที่เก็บไฟล์ทั้งหมดเอาไว้ (ในที่นี้คือโฟลเดอร์ say_hello) โดยไฟล์ที่ซิปนั้นจะเก็บโฟลเดอร์ที่เก็บไฟล์ที่ใช้งานของแอปพลิเคชันทั้งหมดอยู่ภายใน

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ปัจจุบันการใช้งานเทคโนโลยีอินเทอร์เน็ตความเร็วสูงมีจำนวนเพิ่มขึ้นอย่างต่อเนื่อง ประกอบกับการที่เว็บเบราว์เซอร์มีความสามารถมากขึ้น จึงส่งผลให้นักพัฒนาสามารถพัฒนาเว็บแอปพลิเคชันที่มีประสิทธิภาพสูงได้ และแนวคิดของเว็บโอเอสหรือระบบปฏิบัติการที่ทำงานบนเว็บเบราว์เซอร์ก็ได้รับความสนใจจากนักพัฒนามากขึ้น

โรซ่าโอเอสนั้นเป็นเว็บโอเอส ซึ่งหมายถึง เว็บแอปพลิเคชันที่มีการทำงานพื้นฐานและลักษณะของส่วนติดต่อกับผู้ใช้ที่คล้ายกับระบบปฏิบัติการบนเดสก์ท็อป การพัฒนาโรซ่าโอเอสนี้มีจุดประสงค์หลักเพื่อนำไปใช้งานภายในองค์กร เพื่อให้เกิดความปลอดภัยทางด้านข้อมูลภายในองค์กร และเป็นต้นแบบในการนำไปพัฒนาต่อยอดต่อไป ซึ่งระบบการทำงานภายในโรซ่าโอเอสนั้นประกอบไปด้วย ระบบจัดการข้อมูลของผู้ใช้ ระบบจัดการแอปพลิเคชันสำหรับผู้ใช้ และระบบสำหรับผู้ดูแลระบบ โดยในการใช้งานนั้นระบบจะจัดสรรพื้นที่การใช้งานส่วนตัวบนเว็บให้กับผู้ใช้ โดยผู้ดูแลระบบสามารถปรับแต่งค่าการใช้งานต่างๆของระบบให้เหมาะสมกับการใช้งานขององค์กรของตนได้ นอกจากนี้เพื่อให้นำไปพัฒนาต่อยอดสามารถทำได้อย่างสะดวก ระบบจึงได้จัดเตรียม API ไว้สำหรับนักพัฒนาอีกด้วย โดยในการดำเนินงานปัญหาพิเศษฉบับนี้ได้มีการศึกษาเฟรมเวิร์ก CodeIgniter, jQuery และ jQuery UI เพื่อนำมาช่วยในการพัฒนาระบบ ทำให้การพัฒนา มีความเป็นระบบระเบียบและมีความสะดวกรวดเร็วมากขึ้น เนื่องจากผู้พัฒนาไม่จำเป็นต้องสร้างฟังก์ชันขึ้นมาใช้งานเองทั้งหมด จากการดำเนินงานปัญหาพิเศษฉบับนี้ ผลลัพธ์ที่ได้คือเว็บโอเอสที่ผู้ใช้สามารถนำไปใช้งานเบื้องต้นภายในองค์กร และสามารถนำไปพัฒนาต่อยอดได้

5.2 ข้อเสนอแนะ

เนื่องจากระบบปฏิบัติการมีขอบเขตความสามารถในการทำงานที่กว้างและหลากหลาย แต่ในการดำเนินงานปัญหาพิเศษฉบับนี้นั้นมีข้อจำกัดทางด้านเวลา จึงทำให้ไม่สามารถพัฒนาระบบให้มีความสามารถครบถ้วนดังเช่นระบบปฏิบัติการทั่วไปได้ แต่อย่างไรก็ตาม โรซ่าโอเอสซึ่งเป็นผลลัพธ์จากการดำเนินงานปัญหาพิเศษฉบับนี้ก็มีมีความสามารถในการทำงานขั้นพื้นฐาน และผู้พัฒนาได้จัดเตรียมองค์ประกอบต่างๆสำหรับการนำระบบไปพัฒนาต่อยอดไว้แล้ว โดยผู้พัฒนา มีข้อเสนอแนะสำหรับผู้สนใจจะนำระบบพัฒนาต่อยอด ดังนี้

- 1) พัฒนาส่วนทดสอบระบบ (Sandbox) สำหรับผู้ดูแลระบบ เพื่อสามารถศึกษาผลกระทบของการปรับแต่งค่าการใช้งานระบบ การติดตั้ง และยกเลิกการติดตั้งแอปพลิเคชันภายในระบบ ก่อนดำเนินการกับระบบจริง เพื่อดูว่าการดำเนินการใดๆ นั้นส่งผลกระทบในทางที่ก่อให้เกิดข้อผิดพลาดหรือความเสียหายภายในระบบหรือไม่
- 2) ในด้านความปลอดภัยของระบบ เนื่องจากระบบการจัดเก็บไฟล์ของโรซ่าไอเอส ณ ปัจจุบันนั้นยังมีจุดอ่อนทางด้านความปลอดภัย คือ อาจถูกโจมตีได้ด้วยวิธี Forceful browsing หากผู้โจมตีทราบที่อยู่ของไฟล์บนเซิร์ฟเวอร์ ดังนั้นจึงควรพัฒนาระบบจัดเก็บไฟล์แบบใหม่ที่มีความปลอดภัยกว่า และสามารถป้องกันไม่ให้ผู้ใช้สามารถเข้าถึงไฟล์ได้โดยตรง
- 3) พัฒนาระบบแอปพลิเคชันสโตร์ (App Store) เพื่อเป็นช่องทางสำหรับนักพัฒนาในการแจกจ่ายแอปพลิเคชันที่พัฒนาขึ้นให้กับผู้ใช้คนอื่นๆ สามารถดาวน์โหลดไปติดตั้งใช้งานได้
- 4) ในด้านความสามารถเพิ่มเติมของระบบ อาจพัฒนาระบบการแชร์ไฟล์ข้อมูลระหว่างผู้ใช้หรือกลุ่มของผู้ใช้ และพัฒนาให้ระบบสามารถบันทึกข้อมูลการใช้งานล่าสุดของผู้ใช้ไว้ เพื่อให้ผู้ใช้สามารถทำงานที่ค้างไว้ต่อเนื่องได้ทันทีเมื่อล็อกอินเข้าสู่ระบบในครั้งต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- Ali Sajedi, Mehregan Mahdavi, Amir pour shir mohammadi and Minoos Monajjemi nejad, “*Fundamental Usability Guidelines for User Interface Design*”, International Conference on Computational Sciences and Its Applications ICCSA, 2008.
- Feetu Nyrhinen and Tommi Mikkonen, “*Web Browser as a Uniform Application Platform: How Far Are We?*”, 35th Euromicro Conference on Software Engineering and Advanced Applications, 2009.
- eyeOS Company, “*EyeOS 2.0 Developer Manual*”, 2010.
- [Online].Available: www.iamatechie.com/web-based-operating-system-eyeos/, “*Web based operating system : eyeOS*”, 2011.
- [Online].Available: www.youos.com/html/static/manifesto/what.html, “*What is YouOS?*”, 2011.
- [Online].Available: computer.howstuffworks.com/web-operating-system.htm/printable, “*How Web Operating Systems Work*”, 2011.
- [Online].Available: www.eyeos.org/, “*eyeOS - Web Desktop, Cloud Computing Operating System and Web Office*”, 2011.
- [Online].Available: www.jquery.com/, “*jQuery: The Write Less, Do More, JavaScript Library*”, 2011.
- [Online].Available: www.jqueryui.com/, “*jQuery UI is the official jQuery user interface library. It provides interactions, widgets, effects, and theming for creating Rich Internet Applications.*”, 2011.
- [Online].Available: www.codeigniter.com/, “*CodeIgniter - Open source PHP web application framework*”, 2011.