

การพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์
ในรูปแบบแฟ้มเจพีค



นายอัศวิน วงษ์แก่นคำ

ศูนย์วิทยทรัพยากร
วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2552

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DEVELOPMENT OF A SUPPORTING SYSTEM FOR ONLINE PDF VIEWER
IN JPEG FILE FORMAT



Mr.Ussawin Vongkancom

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2009

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์

การพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผล
แฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจพีค

โดย

นายอัศวิน วงษ์แก่นคำ

สาขาวิชา

วิทยาศาสตร์คอมพิวเตอร์

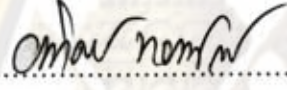
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

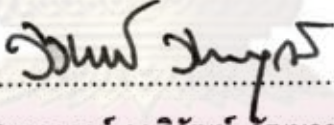
รองศาสตราจารย์ ดร. วิวัฒน์ วัฒนาวุฒิ

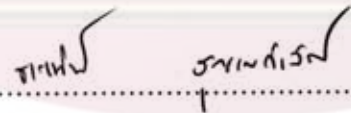
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาโทบัณฑิต

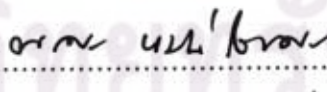

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศhiratวงศ์)

คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ)


..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)


..... กรรมการ
(รองศาสตราจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์)


..... กรรมการ
(รองศาสตราจารย์ ดร.พรศิริ นมื่นไชยศรี)


..... กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.นวลวรรณ สุนทรกีซ)

อัศวิน วงษ์แก่นคำ : การพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจเพิก. (DEVELOPMENT OF A SUPPORTING SYSTEM FOR ONLINE PDF VIEWER IN JPEG FILE FORMAT) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : รศ. ดร. วิวัฒน์ วัฒนาวุฒิ, 61 หน้า.

งานวิจัยนี้ได้ศึกษาและพัฒนาระบบการแสดงผลข้อมูลของแฟ้มพีดีเอฟโดยการแปลงแฟ้มพีดีเอฟเป็นแฟ้มรูปภาพ ซึ่งได้เลือกใช้แฟ้มชนิดเจเพิกเพื่อนำมาแสดงให้กับผู้ใช้ผ่านระบบอินเทอร์เน็ตโดยแสดงผลบนเว็บเบราว์เซอร์ต่าง ๆ ได้ ซึ่งออกแบบการใช้งานด้วยโปรแกรมภาษาจาวาสคริปต์ บนส่วนต่อประสานผู้ใช้มีความสามารถในการย่อขยายและค้นหาข้อมูลภายในแฟ้มพีดีเอฟ ในการพัฒนาได้ใช้ ImageMagick มาช่วยแปลงเป็นแฟ้มเจเพิก และใช้ XPDF สำหรับการแปลงเป็นข้อมูลตัวอักษร

จากการทดสอบการค้นหาค่ามีความถูกต้องประมาณ 98.69% ซึ่งสามารถนำมาทดแทนการใช้โปรแกรมประยุกต์ที่ต้องอ่านข้อมูลจากแฟ้มพีดีเอฟโดยตรงได้ ทั้งนี้จากการทดสอบเปิดเอกสารวิทยานิพนธ์ที่ได้จากคลังปัญญาจุฬา ฯ เพื่อประเทศไทยที่อยู่ในรูปแบบแฟ้มพีดีเอฟจำนวน 30 ฉบับ ใช้เวลาในการแปลงข้อมูลเป็นแฟ้มเจเพิกต่อหนึ่งหน้าโดยเฉลี่ยแล้วประมาณ 1 วินาที ดังนั้นจากการใช้งานสามารถที่จะเปิดแฟ้มพีดีเอฟได้ทันทีจึงสามารถนำมาใช้เพื่อช่วยแก้ไขปัญหาในการเปิดแฟ้มพีดีเอฟแบบออนไลน์ที่ต้องดาวน์โหลดแฟ้มพีดีเอฟมาที่เครื่องผู้ใช้ทั้งหมดก่อนจึงเปิดดูข้อมูลได้ อีกทั้งยังสามารถพัฒนาให้มีการจำกัดการอ่านเอกสารของผู้ใช้หรือสามารถเปิดเอกสารได้บางส่วน

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชาวิศวกรรมคอมพิวเตอร์

ลายมือชื่อ นิสิต..... อัศวิน วงษ์แก่นคำ

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์

ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก..... วิวัฒน์ วัฒนาวุฒิ

ปีการศึกษา 2552

4971491021 : MAJOR COMPUTER SCIENCE

KEYWORDS : Online PDF Browser / Digital Library

USSAWIN VONGKANCOM : DEVELOPMENT OF A SUPPORTING SYSTEM
FOR ONLINE PDF VIEWER IN JPEG FILE FORMAT. THESIS ADVISOR :
ASSOC. PROF. WIWAT VATANAWOOD, Ph.D., 61 pp.

This research proposes a online browser system of PDF files by converting PDF files to JPEG image files in order to show users via the web browser. The Javascript language programming is mainly used to implement this feature. To manipulate the displayed web page, the user can zoom in and out, and search any line of the document in PDF files using keyword. ImageMagick is used in converting each page of PDF file to JPEG file, and XPDF is used to convert the file to alphabetical tokens.

The test cases are conducted to ensure the keyword searching. It yields the accuracy of 98.69% which can be used as a replacement for the application that must directly read information from PDF files. More than 30 theses from Chulalongkorn University Intellectual Repository (CUIR) have been tested. It requires approximately one minute for converting the information to one page of JPEG file. Therefore, it is possible to open PDF file immediately and help solving the problem of opening PDF file online that requires all of the PDF files to be downloaded before it can be opened. The pages of the PDF file would be exclusively limited to the specified group of users if needed and the page access and viewing are controlled to authorized group of user, as well.

Department : ...Computer Engineering.....

Student's Signature

Field of Study : ...Computer Science.....

Advisor's Signature

Academic Year : ...2009.....

USSAWIN VONGKANCOM
Wiwat Vatana Wood

จุฬาลงกรณ์มหาวิทยาลัย

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความอนุเคราะห์อย่างยิ่งของรองศาสตราจารย์ ดร. วิวัฒน์ วัฒนาวุฒิ อาจารย์ที่ปรึกษาซึ่งได้ให้ความรู้ประสิทธิภาพวิชา แนะนำแนวทางการวิจัย ให้กำลังใจ และให้การสนับสนุนเป็นอย่างดี จนทำให้การวิจัยในครั้งนี้สำเร็จออกมาด้วยดี

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์, รองศาสตราจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์, รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี และผู้ช่วยศาสตราจารย์ ดร. นवलวรรณ สุนทรวิเศษ กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลา ให้คำแนะนำ ตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

ท้ายที่สุด ผู้เสนอวิทยานิพนธ์ขอขอบคุณเพื่อน ๆ ทุก ๆ คน รวมทั้งครอบครัว เพื่อนร่วมงาน และผู้บังคับบัญชาในสายงาน ที่คอยติดตาม ให้กำลังใจและสนับสนุน รวมถึงท่านอื่น ๆ ที่ได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนช่วยให้วิทยานิพนธ์สำเร็จได้ด้วยดี

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฌ
สารบัญภาพ.....	ญ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	2
1.3 ขอบเขตของการวิจัย.....	2
1.4 วิธีดำเนินการวิจัย.....	2
1.5 ลำดับขั้นตอนในการเสนอผลการวิจัย.....	3
1.6 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์.....	3
1.7 ประโยชน์ที่จะได้รับ.....	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	4
2.1 แฟ้มพีดีเอฟ (PDF) [1].....	4
2.2 แฟ้มข้อมูลภาพชนิดเจเพ็ก (JPEG - Joint Photographic Experts Group) [3].....	7
2.3 ImageMagick [4].....	11
2.4 จาวาสคริปต์ [5].....	11
2.5 เอแจ็กซ์ (AJAX - Asynchronous JavaScript And XML) [6].....	12
2.6 XMLHttpRequest [9].....	15
2.7 พีเอชพี (PHP) [11].....	16
บทที่ 3 การวิเคราะห์ออกแบบสถาปัตยกรรมระบบ.....	19
3.1 การวิเคราะห์ความต้องการ.....	19
3.2 การวิเคราะห์ระบบ.....	20

	หน้า
3.3 การออกแบบระบบ.....	20
บทที่ 4 การพัฒนาระบบ.....	31
4.1 ขั้นตอนการพัฒนาระบบ.....	31
4.2 ตัวอย่างส่วนต่อประสานผู้ใช้ (User Interface).....	32
4.3 การจัดเก็บแฟ้มของโปรแกรมที่พัฒนา.....	33
4.4 การทดสอบการทำงานของระบบ.....	35
4.5 การวัดประสิทธิภาพของระบบ.....	38
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....	41
5.1 สรุปผลการวิจัย.....	41
5.2 ข้อจำกัด.....	42
5.3 ปัญหาในการใช้งาน.....	42
5.3 แนวทางการวิจัยต่อ.....	42
รายการอ้างอิง.....	43
ภาคผนวก.....	44
ภาคผนวก ก. คำอธิบายยูสเคส.....	45
ภาคผนวก ข. ตัวอย่างการใช้งานระบบบนเว็บเบราว์เซอร์ต่าง ๆ	50
ภาคผนวก ค. บทความวิชาการ.....	54
ประวัติผู้เขียนวิทยานิพนธ์.....	61

สารบัญตาราง

	หน้า
ตารางที่ 4.1 ผลการทดสอบการค้นหาคำ.....	37
ตารางที่ 4.2 ผลการวัดความเร็วในการแปลงเอกสาร.....	40
ตารางที่ ก.1 คำอธิบายยูสเคส Insert Form.....	45
ตารางที่ ก.2 คำอธิบายยูสเคส Register System.....	45
ตารางที่ ก.3 คำอธิบายยูสเคส Login System.....	46
ตารางที่ ก.4 คำอธิบายยูสเคส View PDF.....	46
ตารางที่ ก.5 คำอธิบายยูสเคส Count Page.....	47
ตารางที่ ก.6 คำอธิบายยูสเคส Render Page.....	47
ตารางที่ ก.7 คำอธิบายยูสเคส Scroll Page.....	48
ตารางที่ ก.8 คำอธิบายยูสเคส Zoom PDF Page.....	48
ตารางที่ ก.9 คำอธิบายยูสเคส Search Page.....	49
ตารางที่ ก.10 คำอธิบายยูสเคส Search Data.....	49



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

	หน้า
รูปที่ 2.1 การบวนการสร้างแฟ้มพีดีเอฟโดยวิธีพีดีเอฟไรเตอร์.....	4
รูปที่ 2.2 กระบวนการสร้างแฟ้มพีดีเอฟโดยวิธีดิสทิลเลอร์.....	5
รูปที่ 2.3 ภาพรวมขั้นตอนการทำงานของกริปปี้อัดเจเพ็ก.....	7
รูปที่ 2.4 สัมประสิทธิ์ความถี่ขนาด 8 x 8 หลังผ่านการแปลงดีซีที.....	9
รูปที่ 2.5 การทำควอนไทซ์ของมาตรฐานเจเพ็ก.....	10
รูปที่ 2.6 การสแกนแบบซิกแซกของเมทริกซ์ขนาด 8 x 8.....	10
รูปที่ 2.7 ตัวอย่างการใช้คำสั่ง Imagick.....	11
รูปที่ 2.8 เปรียบเทียบการทำงานระหว่างเว็บแอปพลิเคชันแบบดั้งเดิม กับแบบที่ใช้เอเจ็กซ์.....	14
รูปที่ 2.9 เปรียบเทียบการติดต่อสื่อสารระหว่างเว็บแอปพลิเคชันแบบดั้งเดิมกับแบบที่ใช้ เอเจ็กซ์.....	15
รูปที่ 2.10 การเรียกใช้ XMLHttpRequest ใน IE7 และเว็บเบราว์เซอร์อื่น.....	16
รูปที่ 2.11 การเรียกใช้ XMLHttpRequest ใน IE5.x และ IE6.....	16
รูปที่ 2.12 การรวมการเรียกใช้ XMLHttpRequest กับทุกเว็บเบราว์เซอร์.....	16
รูปที่ 2.13 การใช้งานภาษาพีเอชพี.....	17
รูปที่ 2.14 การใช้งานภาษาพีเอชพีโดยไม่กำหนดตัวแปร.....	18
รูปที่ 3.1 แผนภาพยูสเคสการวิเคราะห์ระบบ.....	20
รูปที่ 3.2 การออกแบบขั้นตอนการทำงานของระบบ.....	21
รูปที่ 3.3 การเรียกใช้งานจาวาสคริปต์.....	22
รูปที่ 3.4 การใช้เอเจ็กซ์เพื่อขอจำนวนหน้าพีดีเอฟ.....	23
รูปที่ 3.5 ฟังก์ชันการนับจำนวนหน้าในแฟ้มพีดีเอฟ.....	23
รูปที่ 3.6 คำสั่งแปลงข้อมูลแฟ้มพีดีเอฟเป็นแฟ้มเจเพ็ก.....	24
รูปที่ 3.7 การออกแบบขั้นตอนการย่อขยายเอกสารระบบแสดงผลแฟ้มพีดีเอฟ.....	26
รูปที่ 3.8 การออกแบบขั้นตอนการค้นหาข้อมูลภายในแฟ้มพีดีเอฟ.....	27
รูปที่ 3.9 คำสั่งการค้นหาค่าภายในแฟ้มพีดีเอฟ.....	28
รูปที่ 3.10 คำสั่งการค้นหาหัวข้อหลักภายในแฟ้มพีดีเอฟ.....	29
รูปที่ 3.11 แผนภาพสถาปัตยกรรมของระบบ.....	30

	หน้า
รูปที่ 4.1 ส่วนต่อประสานผู้ใช้ของระบบลงทะเบียนและเข้าสู่ระบบ.....	32
รูปที่ 4.2 รายละเอียดวิทยานิพนธ์ตำแหน่งที่มีการเปลี่ยนจุดเชื่อมโยง.....	33
รูปที่ 4.3 ส่วนต่อประสานผู้ใช้ระบบแสดงผลแฟ้มพีดีเอฟ.....	33
รูปที่ 4.4 โครงสร้างของไฟล์เดอริในการจัดเก็บโปรแกรมของระบบ.....	34
รูปที่ 4.5 ตัวอย่างการเปิดแฟ้มพีดีเอฟโดยใช้โปรแกรมประยุกต์.....	35
รูปที่ 4.6 ตัวอย่างการเปิดแฟ้มพีดีเอฟโดยใช้ระบบที่พัฒนาขึ้น.....	35
รูปที่ 4.7 ผลการวัดประสิทธิภาพของระบบ.....	39
รูปที่ 4.8 ผลการวัดประสิทธิภาพของระบบในขณะค้นหาข้อมูล.....	39
รูปที่ ข.1 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Internet Explorer 6.....	51
รูปที่ ข.2 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Internet Explorer 7.....	51
รูปที่ ข.3 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Mozilla Firefox.....	52
รูปที่ ข.4 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Apple Safari.....	52
รูปที่ ข.5 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Opera.....	53

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เอกสารอิเล็กทรอนิกส์แบบพีดีเอฟ (PDF - Portable Document Format) เริ่มมีความนิยมแพร่หลายมากขึ้นเนื่องด้วยความสะดวกในหลายประการเช่น การจัดเก็บ เพราะเป็นเพียงแฟ้มข้อมูลไม่ต้องใช้เนื้อที่เหมือนหนังสือจริง ง่ายต่อการเสาะหาเพราะสามารถดาวน์โหลดออนไลน์ผ่านทางอินเทอร์เน็ตได้ การพกพาอุปกรณ์อิเล็กทรอนิกส์ปัจจุบันมีอุปกรณ์พกพาหลายประเภทสามารถแสดงผลเอกสารอิเล็กทรอนิกส์ได้อย่างเช่น โทรศัพท์มือถือ พีดีเอ (PDA) คอมพิวเตอร์พกพา (Laptop) ทำให้สามารถอ่านเอกสารได้ในทุก ๆ ที่ แต่การที่จะดาวน์โหลดเอกสารมานั้น อาจจะเป็นการลำบากถ้าหากเอกสารนั้นมากขนาดของข้อมูลที่ใหญ่มาก ทำให้ต้องเสียเวลาดาวน์โหลด และเมื่อดาวน์โหลดมาแล้ว เอกสารนั้นอาจจะมีเนื้อหาที่ไม่ตรงตามความต้องการของผู้ใช้ ทำให้เสียเวลาโดยใช่เหตุ

ระบบคลังเอกสารของระบบ อนุญาตให้มีการอ่านเอกสารบางประเภทเช่นเอกสารงานวิจัย วิทยานิพนธ์นั้นไม่ต้องการให้ผู้ใช้สามารถอ่านข้อมูลในเอกสารได้ทั้งหมด คือต้องการให้อ่านได้เพียงบางส่วนของเอกสารนั้น ๆ หรือต้องการให้อ่านทั้งหมดได้แต่ไม่ต้องการให้สามารถดาวน์โหลดเอกสารนั้นไปได้ แต่เนื่องจากคุณสมบัติของเอกสารอิเล็กทรอนิกส์ หรือในที่นี่จะเรียกว่า “แฟ้มพีดีเอฟ” มีลักษณะเป็นแฟ้มเอกสารที่ได้รวมข้อมูลของเอกสารทุกอย่างไว้ในแฟ้ม ๆ เดียว ซึ่งถ้าผู้ใช้ต้องการอ่านแฟ้มพีดีเอฟที่อยู่ในระบบอินเทอร์เน็ต จะต้องทำการดาวน์โหลดแฟ้มพีดีเอฟมาก่อนทั้งหมดแล้วจึงสามารถเปิดอ่านเอกสารได้ตามที่ได้กล่าวมาแล้ว และการอ่านนั้นจะอ่านข้อมูลในเอกสารได้ทั้งหมดทุกหน้า โดยที่ผู้ที่ให้บริการ หรือเจ้าของแฟ้มพีดีเอฟนั้นไม่สามารถกำจัดการจำนวนหน้าในการอ่านได้เลย

การแสดงผลเนื้อหาของเอกสารสำหรับในงานวิจัยนี้จะกล่าวถึงแฟ้มเอกสารอิเล็กทรอนิกส์แบบ พีดีเอฟเพียงอย่างเดียว ซึ่งจากเหตุผลต่าง ๆ ที่ได้กล่าวมาแล้วนั้น จึงเป็นปัญหาสำคัญในความต้องการจำกัดเนื้อหาในการอ่าน และป้องกันการดาวน์โหลดแฟ้มพีดีเอฟ โดยในปัจจุบันถ้าต้องการให้ผู้ใช้อ่านแฟ้มพีดีเอฟ โดยจำกัดจำนวนเนื้อหาตามที่ต้องการจะต้องทำการสำเนาแฟ้มขึ้นมาใหม่แล้วตัดข้อมูลในแฟ้มให้เหลือเฉพาะส่วนเนื้อหาที่ต้องการให้อ่านได้เท่านั้น ซึ่งวิธีนี้จะเสียเวลาและเสียเนื้อที่ในระบบโดยใช่เหตุและยังทำให้การจัดการแฟ้มเป็นเรื่องยุ่งยากคือ เอกสารเรื่องเดียวแต่มีแฟ้มมากกว่า 1 แฟ้ม

วิทยานิพนธ์นี้จึงได้ศึกษาและพัฒนาวิธีการแสดงข้อมูลภายในแฟ้มพีดีเอฟ ซึ่งได้พบว่ามีวิธีที่สามารถนำแฟ้มพีดีเอฟแปลงเป็นแฟ้มรูปภาพได้ โดยได้เลือกใช้แฟ้มรูปภาพชนิดเจเพ็ก (JPEG) เพื่อให้ได้ภาพที่มีความคล้ายกับต้นฉบับและมีขนาดเล็ก จากนั้นนำแฟ้มเจเพ็กที่ได้นำมาแสดงให้กับผู้ใช้ผ่านระบบอินเทอร์เน็ตโดยแสดงบนเว็บเบราว์เซอร์ที่ออกแบบมาให้เหมือนกับการใช้งานบนโปรแกรมแอปพลิเคชันสำหรับอ่านแฟ้มพีดีเอฟ

1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจเพ็ก

1.3 ขอบเขตของการวิจัย

1. สามารถแสดงเนื้อหาของแฟ้มพีดีเอฟโดยใช้การแสดงผลรูปภาพ
2. สามารถกำหนดให้แสดงเอกสารเฉพาะบางหน้าตามที่ต้องการ หรือแสดงทุกหน้า
3. สามารถเลือกแสดงเฉพาะเนื้อหาส่วนที่เป็นรูปภาพภายในแฟ้มพีดีเอฟได้ โดยแฟ้มพีดีเอฟดังกล่าวต้องสร้างจากข้อมูลที่เป็นตัวอักษร และรูปภาพจริง ไม่ได้สร้างจากการสแกนเป็นรูปภาพ
4. สามารถค้นหาคำ และตำแหน่งของคำสำคัญที่บ่อนภายในเอกสารได้
5. สามารถจัดทำหัวข้อเรื่อง เพื่อแสดงข้อมูลแบบสารบัญ
6. สามารถย่อขยายเอกสารได้ 10 ระดับในช่วง 10 - 800 %
7. การใช้งานระบบสนับสนุนโปรแกรมค้นดูแฟ้มพีดีเอฟแบบออนไลน์ สามารถทำงานผ่านเว็บเบราว์เซอร์ที่ชื่อ Internet Explorer, Mozilla Firefox, Apple Safari, Opera
8. ใช้คลังปัญญาจุฬา ฯ เพื่อประเทศไทยเป็นกรณีทดสอบติดตั้งและใช้งาน
9. พัฒนาระบบลงทะเบียนผู้อ่านให้สามารถตรวจสอบตัวตนและสิทธิการเข้าดูเอกสาร
10. ทดลองใช้งาน และรายงานผลการวัดสมรรถนะ (Performance) ของระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจเพ็ก

1.4 วิธีดำเนินการวิจัย

1. ศึกษาวิธีการแปลงแฟ้มพีดีเอฟเป็นแฟ้มรูปภาพ
2. ศึกษาและออกแบบเว็บแอปพลิเคชันสำหรับแสดงข้อมูลจากแฟ้มเอกสาร
3. ออกแบบขั้นตอนการทำงานร่วมกันระหว่างส่วนแปลงแฟ้มข้อมูล และส่วนการแสดงผล

4. พัฒนาด้านแบบระบบการแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ตามที่ได้ออกแบบไว้
5. ทำการติดตั้ง และทดสอบการทำงานของโปรแกรมที่พัฒนา
6. สรุปผล และประเมินผลการทดสอบ
7. จัดทำเอกสารสรุปงานวิจัย และข้อเสนอแนะ

1.5 ลำดับขั้นตอนในการเสนอผลการวิจัย

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 5 บทดังต่อไปนี้ บทที่ 1 เป็นบทนำซึ่งกล่าวถึง ความ เป็นมาและความสำคัญของปัญหา รวมถึงวัตถุประสงค์ของการวิจัย บทที่ 2 กล่าวถึงทฤษฎีและ งานวิจัยที่เกี่ยวข้องในงานวิจัยนี้ บทที่ 3 กล่าวถึงการออกแบบสถาปัตยกรรมระบบ บทที่ 4 เป็น การพัฒนาระบบสนับสนุน และท้ายสุดคือบทที่ 5 กล่าวถึงสรุปผลการวิจัยและข้อเสนอแนะ

1.6 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตอบรับให้ตีพิมพ์เป็นบทความทางวิชาการในหัวข้อ เรื่อง “ระบบการอ่านเอกสารอิเล็กทรอนิกส์หลากหลายหน้าแบบออนไลน์ (Online Multi-Page Electronic Document Reader System)” โดย อัครวิน วงษ์แก่นคำ, วิวัฒน์ วัฒนาวุฒิ นำเสนอ ในงานประชุมวิชาการ “The 2nd National Conference on Information Technology (NCIT 2008)” ณ โรงแรมฟอร์จูน แกรนด์ เมอร์เคียว กรุงเทพมหานคร ระหว่างวันที่ 6-7 พฤศจิกายน 2551

1.7 ประโยชน์ที่จะได้รับ

สามารถนำไปใช้แสดงเนื้อหาภายในแฟ้มพีดีเอฟได้โดยสามารถกำหนดให้ผู้ใช้สามารถ อ่านเนื้อได้เพียงส่วนใดส่วนหนึ่ง และกรณีที่แฟ้มพีดีเอฟมีจำนวนหน้าและขนาดแฟ้มใหญ่มากผู้ใช้ ไม่จำเป็นต้องดาวน์โหลดแฟ้มพีดีเอฟไปทั้งหมดก็สามารถที่จะอ่านเนื้อได้ทันที

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

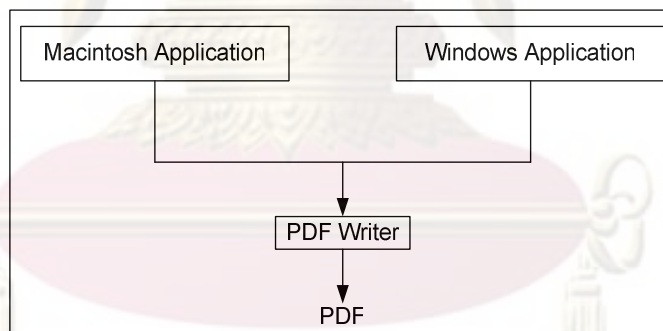
2.1 แฟ้มพีดีเอฟ (PDF) [1]

ก่อนจะเข้าสู่กระบวนการแปลงแฟ้มพีดีเอฟ ควรจะทราบถึงข้อกำหนด และแนวคิดของแฟ้มพีดีเอฟ เช่น วิธีการสร้างแฟ้มพีดีเอฟ โครงสร้างแฟ้มพีดีเอฟ ข้อกำหนดประเภทแบบอักษร ข้อกำหนดการเข้ารหัสของแบบอักษรที่ใช้ในเอกสารพีดีเอฟ

2.1.1 การสร้างแฟ้มพีดีเอฟ

การสร้างเอกสารพีดีเอฟ สามารถสร้างได้ 2 วิธี คือ

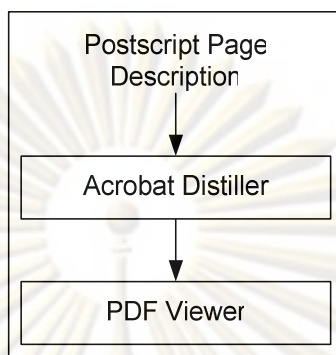
- 1) พีดีเอฟไรเตอร์ (PDF Writer) สร้างแฟ้มพีดีเอฟจากโปรแกรมสำเร็จประยุกต์ที่ผู้ใช้นิยมใช้โดยสั่งให้โปรแกรมสำเร็จประยุกต์นั้นพิมพ์เอกสารผ่านพีดีเอฟไรเตอร์ พีดีเอฟไรเตอร์จะทำการนำรายละเอียดการพิมพ์นั้นมาสร้างเป็นเอกสารพีดีเอฟ ดังรูปที่ 2.1



รูปที่ 2.1 การบวนการสร้างแฟ้มพีดีเอฟโดยวิธีพีดีเอฟไรเตอร์

- 2) ดิสทิลเลอร์ (Distiller) เป็นวิธีการสร้างเอกสารพีดีเอฟโดยจะทำการแปลงจากแฟ้มโพสต์สคริปต์ (Postscript) [2] มาเป็นแฟ้มพีดีเอฟ เอกสารที่ประกอบด้วยข้อความทั่ว ๆ ไปในการสร้างเป็นแฟ้มพีดีเอฟ สามารถทำได้ทั้ง 2 วิธี แต่ในแฟ้มที่มีรูปภาพต้องการคุณภาพของรูป รูปที่จะแสดงได้สวยงาม ควรเลือกใช้วิธีดิสทิลเลอร์ การสร้างแฟ้มพีดีเอฟโดยวิธีนี้จะได้แฟ้มที่มีคุณภาพดีกว่าโดยวิธีพีดีเอฟไรเตอร์ เนื่องจากผู้ใช้จะสร้างแฟ้มให้อยู่ในรูปแบบแฟ้มโพสต์สคริปต์ก่อน แล้วจึง

ใช้ดีสทิลเลอร์ทำการแปลงแฟ้มนั้นเป็นแฟ้มพีดีเอฟ ซึ่งในการสร้างแฟ้มโพสต์สคริปต์นั้น ผู้ใช้สามารถเขียนแฟ้มขึ้นเองโดยใช้คำสั่งภาษาโพสต์สคริปต์



รูปที่ 2.2 กระบวนการสร้างแฟ้มพีดีเอฟโดยวิธีดีสทิลเลอร์

2.1.2 โครงสร้างแฟ้มพีดีเอฟ

ประกอบด้วย 4 ส่วนด้วยกัน ดังนี้

- 1) ส่วนหัว (Header) ใช้ในการระบุว่า แฟ้มพีดีเอฟนี้เป็นไปตามข้อกำหนดรุ่นที่เท่าไร เช่น ที่ได้ระบุในบรรทัดแรกของแฟ้มเป็น %PDF-1.2 ซึ่งหมายถึงโครงสร้างแฟ้มพีดีเอฟนี้เป็นไปตามข้อกำหนดรุ่นที่ 1.2
- 2) ส่วนที่ใช้ในการแสดงเอกสาร (Body) จะประกอบด้วยวัตถุต่าง ๆ ที่ใช้ในการแสดงข้อความ รูปภาพ เสียง ภาพเคลื่อนไหว และข้อมูลอื่น ๆ ในส่วนนี้จะถูกนำมาใช้ในการแสดงแฟ้มพีดีเอฟ วัตถุต่าง ๆ ในส่วนนี้สัมพันธ์กันในลักษณะของโครงสร้างข้อมูลแบบต้นไม้ ข้อมูลบางส่วน เช่นเนื้อหาในแฟ้มแบบอักษรที่ผนวกมากับแฟ้ม โดยปกติจะถูกบีบอัดข้อมูลไว้เพื่อลดขนาดของข้อมูล ในการใช้วัตถุในแฟ้มต้องทำการขยายข้อมูลที่ถูกลบอัดไว้เสียก่อนจึงจะใช้วัตถุต่าง ๆ ในส่วนนี้ได้
- 3) ตารางอ้างอิง (Cross Reference Table) เป็นข้อมูลตารางที่ช่วยให้เข้าใช้วัตถุในเอกสารพีดีเอฟได้โดยตรง เป็นวิธีการเข้าใช้แบบ “Random Access” ทำให้การอ่านข้อมูลในแฟ้มพีดีเอฟทำได้รวดเร็ว ข้อมูลในส่วนนี้จะต้องเริ่มต้นด้วย xref บรรทัดถัดไปจะแสดงถึงจำนวนวัตถุที่มีอยู่ในเอกสารพีดีเอฟ ตัวอักษร f จะแสดงว่าวัตถุนั้นเป็นวัตถุอิสระหรือเป็นวัตถุเริ่มต้นของรายการเชื่อมโยง (Link list)

ตัวอักษร n จะแสดงว่าวัตถุนั้นถูกใช้ในเอกสาร ตัวเลขในแต่ละแถวจะแสดงถึงตำแหน่งเริ่มต้นของวัตถุนั้นในแฟ้มเอกสารพีดีเอฟ

- 4) ส่วนท้าย (Trailer) ในการอ่านแฟ้มพีดีเอฟจะเริ่มอ่านที่ตอนท้ายของแฟ้มเอกสาร เพื่อให้ทราบตำแหน่งเริ่มต้นของตารางอ้างอิงเพื่อที่จะได้นำไปใช้ในการเข้าใช้วัตถุ ข้อมูลในส่วนท้ายนี้จะต้องเริ่มต้นด้วย trailer และปิดท้ายด้วย %%EOF บรรทัดก่อนที่จะระบุ %%EOF จะบอกถึงตำแหน่งเริ่มต้นของตารางอ้างอิงในแฟ้มพีดีเอฟ นอกจากนี้ในส่วนนี้จะมีข้อมูลอื่นๆที่จะบอกให้ทราบถึง จำนวนวัตถุในแฟ้มพีดีเอฟ วัตถุเริ่มต้น รายละเอียดในการสร้างแฟ้มพีดีเอฟ เช่น ชื่อผู้สร้าง คำสำคัญ วันที่ และอื่น ๆ

2.1.3 แบบอักษรในแฟ้มพีดีเอฟ

แบบอักษรเป็นข้อมูลที่สำคัญในการแสดงข้อมูลในแฟ้มพีดีเอฟ ข้อมูลแบบอักษรในแฟ้มพีดีเอฟ เป็นข้อมูลชนิดพจนานุกรมที่เก็บข้อมูลหรือข้อกำหนดของแบบอักษรมัน ๆ เช่น ชนิดของแบบอักษร ชื่อของแบบอักษร การเข้ารหัสอักษร ข้อมูลที่ใช้ในการแสดงแบบอักษร หรือข้อมูลในการแสดงแทนเมื่อไม่มีแบบอักษรมัน ๆ

แบบอักษรที่ใช้ในแฟ้มพีดีเอฟมีอยู่ด้วยกัน 4 ชนิด คือ

- 1) ประเภทที่ 0 (Type0)
- 2) ประเภทที่ 1 (Type1)
- 3) ประเภทที่ 3 (Type3)
- 4) ประเภททรูไทป์ (TrueType)

แบบอักษรประเภทที่ 0 เป็นแบบอักษรที่ถูกออกแบบมาเพื่อสนับสนุนสำหรับอักขระที่มีตัวอักษรเป็นจำนวนมาก เช่น ภาษาจีน ภาษาญี่ปุ่น ภาษาเกาหลี ซึ่งเป็นภาษาที่มีอักขระมากกว่า 256 อักขระหรือภาษาที่เป็นอักขระภาพ การเข้ารหัสตัวอักษรของแบบอักษรมันนี้จะใช้วิธีการพิเศษเพื่อที่จะสามารถเข้ารหัสอักขระจำนวนมากได้ โดยจะเก็บไว้ในข้อมูลชนิดหนึ่งๆ เรียกว่าแผนที่อักษร

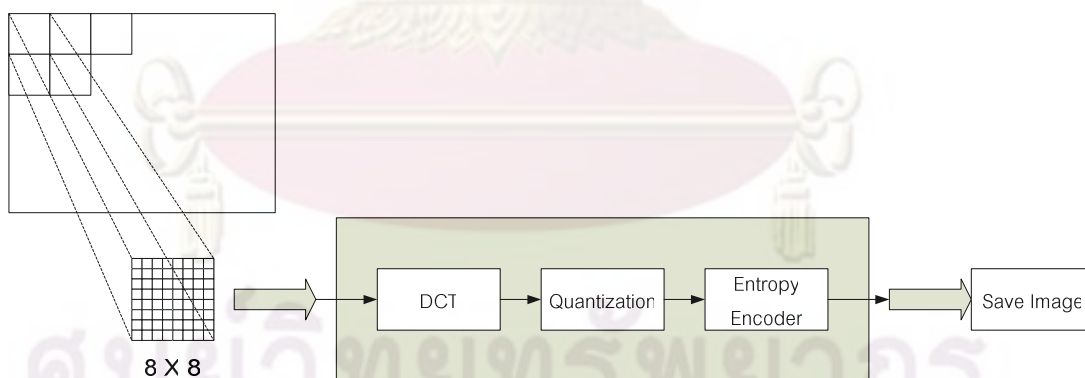
แบบอักษรประเภทที่ 1 มีขนาดเล็ก ให้คุณภาพตัวอักษรดี แม้ว่าจะต้องแสดงตัวอักษรขนาดเล็กบนอุปกรณ์แสดงผลที่มีรายละเอียดในการแสดงผลต่ำ เป็นแบบอักษรที่เหมาะสมที่สุดสำหรับภาษาโพสต์สคริปต์หรือแฟ้มพีดีเอฟ

แบบอักษรประเภทที่ 3 เป็นแบบอักษรที่แตกต่างจากแบบอักษรอื่น ๆ ที่ใช้ในแฟ้มพีดีเอฟ เนื่องจาก แบบอักษรประเภทที่ 3 กำหนดแบบอักษรด้วยตัวเอง ขณะที่พจนานุกรมของแบบอักษรอื่น ๆ เก็บข้อมูลพื้นฐานของแบบอักษรนั้นไว้ แบบอักษรประเภทที่ 3 มีความยืดหยุ่นมากกว่าแบบอักษรประเภทที่ 1 แต่มีคุณภาพในการแสดงด้วยกว่าโดยเฉพาะ ในการแสดงตัวอักษรขนาดเล็กบนอุปกรณ์แสดงผลที่มีรายละเอียดในการแสดงผลต่ำ

แบบอักษรประเภททูไทป์ เป็นแบบอักษรชนิดปรับขนาดได้ ปัจจุบันถูกนำมาใช้เป็นแบบอักษรมาตรฐานระบบปฏิบัติการวินโดวส์ และเป็นมาตรฐานหนึ่งในการแสดงตัวอักษรนี้ในแฟ้มพีดีเอฟ

2.2 แฟ้มข้อมูลภาพชนิดเจเพ็ก (JPEG - Joint Photographic Experts Group) [3]

เจเพ็กเป็นมาตรฐานการบีบอัดภาพนิ่ง (Still Image) ประเภทหนึ่งที่ได้รับการพัฒนาขึ้นโดยกลุ่มผู้เชี่ยวชาญที่มีชื่อเรียกว่า “Joint Photographic Experts Group” มีขีดความสามารถในการบีบอัดได้ทั้งภาพสี และภาพเกรย์สเกล (Grey Scale) การบีบอัดภาพเจเพ็กจัดว่าเป็นวิธีที่มีประสิทธิภาพสูง แฟ้มภาพที่ได้จากการบีบอัดมีขนาดเล็กเมื่อเทียบกับแฟ้มต้นฉบับ อย่างไรก็ตามภาพที่ได้จะไม่เหมือนกับต้นฉบับ 100% แม้คุณภาพของรูปที่เห็นจะใกล้เคียงกับภาพต้นฉบับมาก ดังนั้น การบีบอัดชนิดนี้จัดว่าอยู่ในตระกูลการบีบอัดที่มีการสูญเสีย (Lossy Compression)



รูปที่ 2.3 ภาพรวมขั้นตอนการทำงานของ การบีบอัดเจเพ็ก

การบีบอัดภาพของเจเพ็กมีขั้นตอนการทำงานดังรูปที่ 2.3 โดยขั้นแรกให้นำสัญญาณภาพจากต้นฉบับมาแบ่งออกเป็นบล็อกขนาดเล็ก ๆ แล้วนำไปผ่านกระบวนการแปลงด้วยกรรมวิธีการแปลงโคไซน์ที่ (DCT – Discrete Cosine Transform) เพื่อให้ได้เป็นชุดสัมประสิทธิ์ทางความถี่ จากนั้นให้นำสัมประสิทธิ์ที่ได้ไปผ่านกระบวนการควอนไทซ์ (Quantization) ก่อนจะป้อนเข้าสู่วงจรเข้ารหัสเอนโทรปี (Entropy encoder) เป็นลำดับต่อไป การบวนการทำงานของมาตรฐานเจ

เพ็ญจัดเป็นวิธีการบีบอัดประเภททรานส์ฟอร์มโค้ดดิ้ง (Transform coding) แบบหนึ่ง เหตุผลหลักที่มีการเลือกใช้การแปลงดีซีที เพราะมีขีดความสามารถในการจัดกลุ่มของข้อมูลที่สำคัญให้กระจุกตัวอยู่ในสัมประสิทธิ์เพียงไม่กี่ค่า โดยไม่ต้องการความซับซ้อนในการคำนวณ

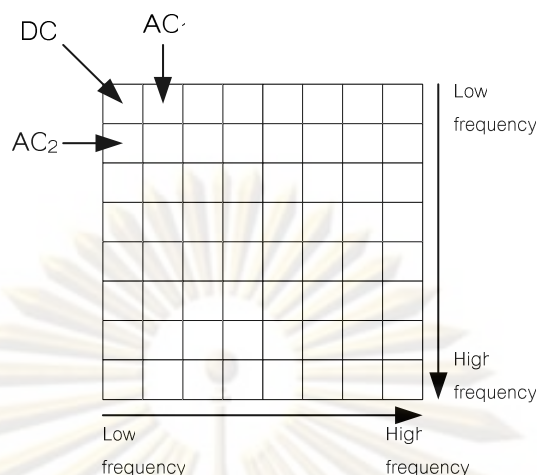
2.2.1 การแปลงดีซีที (DCT)

ข้อมูลภาพที่อยู่ในรูปเมทริกซ์ 2 มิติของพิกเซล (Spatial domain) ถูกนำมาแปลงให้อยู่ในโดเมนความถี่ (Frequency domain) โดยก่อนทำการแปลงดีซีที ข้อมูลภาพจะต้องถูกแบ่งออกเป็นบล็อกเล็ก ๆ ขนาด $N \times N$ ซึ่งตามมาตรฐานเจเพ็ญจะใช้ขนาด 8×8 จากนั้นจึงทำการแปลงดีซีที เพื่อให้ข้อมูลภาพภายในบล็อกแต่ละบล็อกสามารถเขียนแทนฟังก์ชัน $x(i,j)$ โดย $0 \leq i, j \leq 7$ สมการที่ใช้ในการแปลงดีซีทีเป็นดังนี้คือ

$$X(0,0) = \frac{1}{\sqrt{4N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x(i, j) \quad \text{เมื่อ } u, v = 0$$

$$X(u, v) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x(i, j) \cos\left[\frac{(2i+1)u\pi}{2N}\right] \cos\left[\frac{(2j+1)v\pi}{2N}\right] \quad \text{เมื่อ } 1 \leq u, v \leq N-1$$

ผลลัพธ์ที่ได้จากการแปลงดีซีทีจะได้บล็อกของสัมประสิทธิ์ความถี่ขนาดเท่าเดิมคือ $N \times N$ โดย $X(0,0)$ ซึ่งอยู่ ณ ตำแหน่งมุมบนซ้ายสุดของเมทริกซ์จะหมายถึงค่าองค์ประกอบกระแสดตรงของภาพต้นฉบับ นั่นคือ $X(0,0)$ จะแสดงค่าเฉลี่ยความสว่างของพิกเซลภายในบล็อกนั้น ๆ และ $X(u,v)$ ค่าอื่น ๆ แทนองค์ประกอบกระแสดลับของภาพที่ความถี่ต่าง ๆ ซึ่งองค์ประกอบกระแสดลับความถี่ต่ำจะอยู่บริเวณด้านบนซ้ายของเมทริกซ์แล้วเพิ่มความถี่ขึ้นเมื่อเคลื่อนที่ลงมาทางด้านล่างขวาของเมทริกซ์ดังแสดงในรูปที่ 4. โดยองค์ประกอบกระแสดลับของภาพนั้นจะหมายถึงการเปลี่ยนแปลงค่าของสีในแต่ละพิกเซล คือหากค่าสีภายในบล็อกนั้นมีการเปลี่ยนแปลงทีละน้อยต่อเนื่อง ค่าองค์ประกอบกระแสดลับที่ความถี่ต่ำก็จะมีค่ามาก แต่หากค่าสีภายในบล็อกนั้นมีการเปลี่ยนแปลงอย่างรวดเร็ว ค่าองค์ประกอบกระแสดลับที่ความถี่สูงก็จะมีค่ามากขึ้น



รูปที่ 2.4 สัมประสิทธิ์ความถี่ขนาด 8 x 8 หลังผ่านการแปลงดีซีที

2.2.2 การทำควอนไทซ์

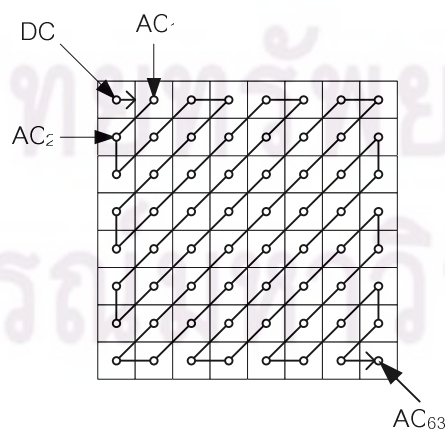
เนื่องจากพลังงานของสัญญาณภาพโดยทั่วไปมักจะประกอบด้วยองค์ประกอบด้วยองค์ประกอบกระแสดตรง และองค์ประกอบกระแสลดที่ความถี่ต่ำเป็นส่วนใหญ่ ดังนั้น สัมประสิทธิ์ของดีซีทีที่คำนวณได้แต่ละตัวจึงมีลำดับความสำคัญไม่เท่ากัน กล่าวคือสัมประสิทธิ์ขององค์ประกอบกระแสดตรงและองค์ประกอบกระแสลดที่ความถี่ต่ำมีความสำคัญสูงกว่า สัมประสิทธิ์ขององค์ประกอบกระแสลดที่ความถี่สูง ด้วยเหตุนี้ สัมประสิทธิ์ที่มีความสำคัญมาก ซึ่งก็คือสัมประสิทธิ์ขององค์ประกอบกระแสดตรงจะได้รับการควอนไทซ์ด้วยระดับความละเอียดสูง ในขณะที่สัมประสิทธิ์ที่ความถี่สูงจะได้รับการควอนไทซ์ค่อนข้างหยาบ การกำหนดระดับความละเอียดของสัมประสิทธิ์แต่ละตัวจึงต้องมีการพิจารณาอย่างถี่ถ้วนโดยคำนึงคุณภาพที่ได้ สำหรับมาตรฐานเจพีคได้กำหนดตารางการทำควอนไทซ์ที่ตายตัวดังแสดงในรูปที่ 2.5 สังเกตว่าตัวเลขที่อยู่ทางมุมบนด้านซ้ายของตารางจะมีขนาดเล็กที่สุด นั่นคือจะได้ความละเอียดสูง ในบริเวณนี้จะใช้กับสัมประสิทธิ์ขององค์ประกอบสัญญาณความถี่ต่ำ และตัวเลขจะมีขนาดเพิ่มขึ้นในทิศทางจากซ้ายไปขวาและบนลงล่าง ซึ่งเป็นการเข้าสู่บริเวณที่มีองค์ประกอบความถี่สูงมากขึ้น

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

รูปที่ 2.5 การทำควอนไทซ์ของมาตรฐานเจเพ็ก

2.2.3 การเข้ารหัสเอนโทรปี (Entropy encoder)

ค่าที่ได้จากขั้นตอนที่แล้วทั้งหมดขนาด 8×8 ทั้ง 64 ค่าจะนำเข้ารหัสเอนโทรปี โดยเริ่มจากการจัดเรียงข้อมูลภายในเมทริกซ์ให้อยู่ในลำดับที่เหมาะสมก่อนกรรมวิธีการจัดเรียงที่เลือกใช้คือ การสแกนแบบซิกแซก (Zig-zag scan) ซึ่งการทำซิกแซกสแกนนั้นจะเป็นการเปลี่ยนค่าเมทริกซ์ 2 มิติ ให้เป็นเวกเตอร์ 1 มิติ นั่นคือเปลี่ยนค่าจากเมทริกซ์ขนาด 8×8 ไปเป็นเวกเตอร์ขนาด 1×64 โดยสัมพันธ์ค่าแรกที่อ่านออกคือ องค์ประกอบกระแสตรง แล้วตามด้วยค่าองค์ประกอบกระแสสลับตัวแรก (AC_1) ซึ่งอยู่มุมบนด้านซ้ายของบล็อก จากนั้นจะอ่านค่าองค์ประกอบกระแสสลับตัวที่สอง (AC_2) แล้วตามด้วยองค์ประกอบกระแสสลับที่ความถี่สูงขึ้นเรื่อย ๆ ตามเส้นที่แสดงในรูปที่ x และค่าท้ายสุดคือ (AC_{63}) เหตุผลที่ทำการเรียงข้อมูลแบบนี้เพื่อที่ต้องการรวมค่าข้อมูลที่เป็น 0 ให้อยู่ติด ๆ กัน เพื่อจะทำให้เกิดประสิทธิภาพในการทำการบีบอัดข้อมูล

รูปที่ 2.6 การสแกนแบบซิกแซกของเมทริกซ์ขนาด 8×8

2.3 ImageMagick [4]

“ImageMagick” คือโปรแกรมเปิด (Open Source) ที่พัฒนาโดย ImageMagick Studio LLC. โดยมีความสามารถ สร้าง แก้ไข ด้วยการประกอบรูปภาพจากจุดหลายๆจุดรวมกัน และโปรแกรมยังสามารถอ่าน แปลง เขียน รูปภาพได้กว่า 90 ชนิด (GIF, JPEG, JPEG-2000, PNG, PDF, PhotoCD, TIFF, and DPX.) สามารถตัดรูปภาพเฉพาะส่วนที่ต้องการ เปลี่ยนสี แสดงผลรูปภาพหลายแบบ (Effects) และเนื่องจากเป็นซอฟต์แวร์เปิด (Open Source) จึงสามารถนำมาใช้งานได้โดยไม่เสียค่าใช้จ่าย

โปรแกรมฟรีจึงสามารถที่จะนำรหัสต้นฉบับ (Source Code) มาใช้ ดัดแปลงแก้ไข และพัฒนาใหม่ได้ และไม่จำกัดเรื่องของระบบปฏิบัติการที่ใช้ ซึ่งสามารถทำงานได้บนเครื่องแม่ข่าย

Imagick เป็นส่วนต่อขยาย (Extension) ของโปรแกรมภาษา PHP โดยจะสามารถนำมาใช้ได้บนเครื่องแม่ข่ายได้ ซึ่ง Imagick มีข้อดีต่างกับ ImageMagik คือมีลักษณะการทำงานแบบ OOP ทำให้ใช้งานง่าย มีความเร็วในการใช้งานเร็วกว่าเนื่องจากไม่ต้องทำงานผ่านคำสั่ง exec ดังรูปที่ 2.7

```
<?php
$im = new Imagick('images/original.jpg');
$im->thumbnailImage(200, null);
$im->borderImage('red', 5, 5);
?>
```

รูปที่ 2.7 ตัวอย่างการใช้คำสั่ง Imagick

2.4 จาวาสคริปต์ [5]

จาวาสคริปต์ (JavaScript) เป็นภาษาสคริปต์ ที่มีลักษณะการเขียนแบบโปรโตไทป์ (Prototyped-based Programming) ส่วนมากใช้ในหน้าเว็บเพื่อประมวลผลข้อมูลที่ฝั่งของผู้ใช้งาน แต่ก็ยังมีใช้เพื่อเพิ่มเติมความสามารถในการเขียนสคริปต์โดยฝังอยู่ในโปรแกรมอื่นๆ

ขั้น ไมโครซิสเต็มส์เป็นเจ้าของเครื่องหมายการค้า "JavaScript" โดยมันถูกนำไปใช้ภายใต้สัญญาอนุญาตเพื่อการพัฒนาเทคโนโลยีโดย เน็ตสเคป และมุลนิธิมอซิลลา

เริ่มพัฒนาโดย Brendan Eich พนักงานบริษัทเน็ตสเคป โดยขณะนั้นจาวาสคริปต์ใช้ชื่อว่า โมคา และภายหลังได้เปลี่ยนชื่อมาเป็น ไลฟ์สคริปต์ และเป็น จาวาสคริปต์ในปัจจุบัน รูปแบบการเขียนภาษาที่ใช้ คล้ายคลึงกับภาษาซี รุ่นล่าสุดของจาวาสคริปต์คือ 2.0 ซึ่งตรงกับมาตรฐานของ ECMAScript

ภาษาจาวาสคริปต์ไม่มีความสัมพันธ์กับ ภาษาจาวา (Java) และ เจสคริปต์ (JScript) แต่อย่างใด ยกเว้นแต่โครงสร้างภาษาที่มีลักษณะคล้ายคลึงกัน เนื่องมาจากได้รับการพัฒนาต่อมาจากภาษาซี และมีชื่อที่คล้ายคลึงกันเท่านั้น

สำหรับเจสคริปต์ (JScript) หลังจากที่จาวาสคริปต์ประสบความสำเร็จ โดยมีเว็บเบราว์เซอร์จากหลายๆ บริษัทนำมาใช้งาน ทางไมโครซอฟท์จึงได้พัฒนาภาษาโปรแกรมที่ทำงานในลักษณะคล้ายคลึงกับจาวาสคริปต์ขึ้น และตั้งชื่อว่าเจสคริปต์ ซึ่งทำงานได้กับเบราว์เซอร์อินเทอร์เน็ตเอกซ์พลอเรอร์ (Internet Explorer) เท่านั้น เริ่มใช้ครั้งแรกใน อินเทอร์เน็ตเอกซ์พลอเรอร์ 3.0 เมื่อ สิงหาคม พ.ศ. 2539 โดยสร้างตามมาตรฐาน ECMA 262

จาวาสคริปต์ เป็นภาษาในรูปแบบของภาษาโปรแกรมแบบโปรโตไทป์ โดยมีโครงสร้างของภาษาและไวยากรณ์อยู่บนพื้นฐานของภาษาซี ซึ่งปัจจุบันมีการใช้จาวาสคริปต์ที่ฝังอยู่ในเว็บเบราว์เซอร์ในหลายรูปแบบ เช่น ใช้เพื่อสร้างเนื้อหาที่เปลี่ยนแปลงเสมอภายในเว็บเพจ, ใช้เพื่อตรวจสอบความถูกต้องของข้อมูลที่ใช้กรอกก่อนนำเข้าระบบ, ใช้เพื่อเข้าถึงข้อมูลที่อยู่ภายใต้โครงสร้างแบบ Document Object Model (DOM) [8] ต้น

นอกจากนี้จาวาสคริปต์ยังถูกฝังอยู่ในแอปพลิเคชันต่างๆ นอกเหนือจากเว็บเบราว์เซอร์ได้ อีกด้วย เช่น widget ของ ยาฮู! เป็นต้น โดยรวมแล้วจาวาสคริปต์ถูกใช้เพื่อให้นักพัฒนาโปรแกรมสามารถเขียนสคริปต์เพื่อสร้างคุณสมบัติพิเศษต่างๆ เพิ่มเติมจากที่มีอยู่บนแอปพลิเคชันดั้งเดิม

โปรแกรมใดๆ ที่สนับสนุนจาวาสคริปต์จะมีตัวขับเคลื่อนจาวาสคริปต์ (JavaScript Engine) ของตัวเอง เพื่อเรียกใช้งานโครงสร้างเชิงวัตถุของโปรแกรมหรือแอปพลิเคชันนั้นๆ

2.5 เอแจ็กซ์ (AJAX - Asynchronous JavaScript And XML) [6]

เป็นเทคนิคในการพัฒนาเว็บแอปพลิเคชันเพื่อให้มีความสามารถในการโต้ตอบระหว่างผู้ใช้ได้ดีขึ้น โดยการรับส่งข้อมูลในเบื้องหลัง ทำให้หน้าเว็บที่ใช้งานอยู่ไม่จำเป็นต้องทำการโหลด

ข้อมูลใหม่ทุกครั้งที่มีการเปลี่ยนแปลง ซึ่งช่วยทำให้เพิ่มการตอบสนอง ความรวดเร็ว และการใช้งานโดยรวม

เอแจ็กซีไม่ใช่เทคโนโลยีใหม่ แต่เป็นเทคนิคที่ได้ใช้เทคโนโลยีหลายอย่างที่มีอยู่มารวมกันดังต่อไปนี้

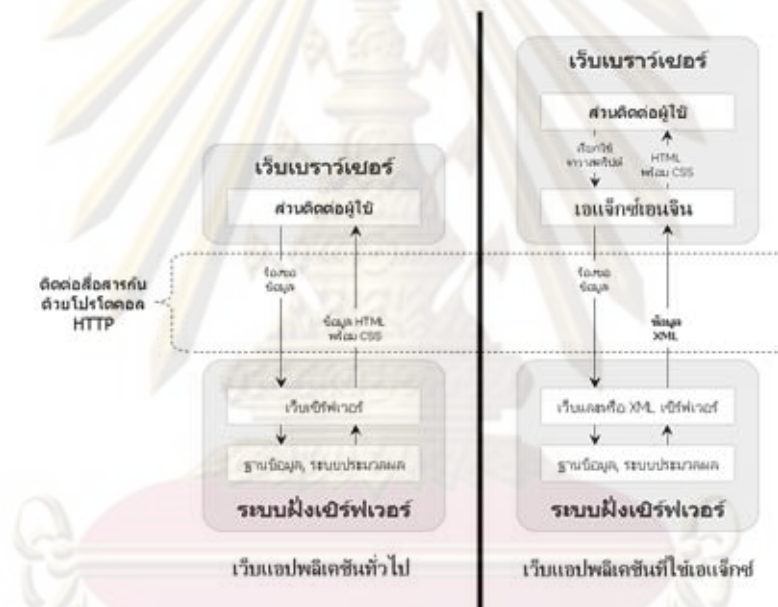
1. XHTML (หรือ HTML) และ CSS [7] สำหรับแสดงผลพีธและรูปแบบของข้อมูล
2. จาวาสคริปต์ สำหรับติดต่อและเข้าถึง Document Object Model (DOM) ใช้ในการแสดงข้อมูลที่มีการเปลี่ยนแปลงหรือโต้ตอบกับผู้ใช้
3. XMLHttpRequest ทำหน้าที่แลกเปลี่ยนข้อมูล Asynchronously กับเว็บเซิร์ฟเวอร์
4. XML เป็นรูปแบบของข้อมูลในการแลกเปลี่ยนระหว่างเว็บเซิร์ฟเวอร์กับเบราว์เซอร์ สำหรับข้อมูลในรูปแบบอื่นก็สามารถใช้ได้เช่นกันไม่ว่าจะเป็น HTML, JSON, EBML, หรือ เฟลนเท็กซ์

Jeese Jams Garrett เป็นผู้ที่ได้บัญญัติคำว่า เอแจ็กซี ขึ้นเมื่อปีพ.ศ. 2548 ซึ่งนี่ก็ขึ้นได้ระหว่างที่เขากำลังอาบน้ำ เพื่อหาคำสั้นๆ สำหรับอธิบายให้ลูกค้าของเขาทราบเกี่ยวกับเทคโนโลยีต่างๆ ที่ต้องการจะนำเสนอ

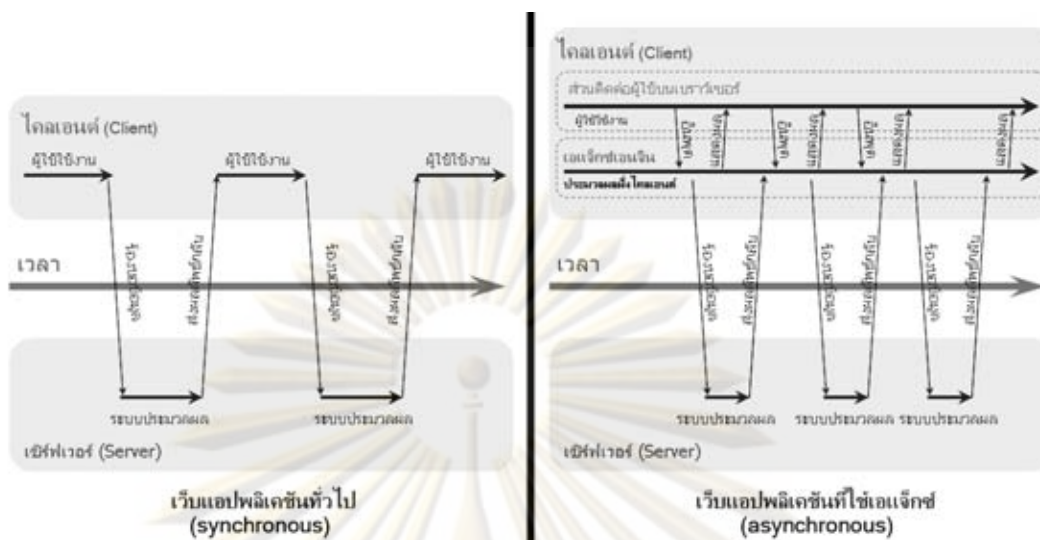
เอแจ็กซีไม่ได้เป็นเทคโนโลยีหรือภาษาโปรแกรมชนิดใหม่ แต่เป็นการรวมกลุ่มของเทคโนโลยีที่มีใช้อยู่แล้วดังที่กล่าวข้างต้น โดยวิวัฒนาการของเอแจ็กซีเริ่มต้นเมื่อปี ค.ศ. 2002 ไมโครซอฟท์ได้ทำการคิดค้น XMLHttpRequest (XHR) ขึ้นมาเพื่อเป็นทางเลือกในการเขียนโปรแกรมบนเว็บเพจ เพื่อใช้ติดต่อกับเว็บเซิร์ฟเวอร์ในการแลกเปลี่ยนข้อมูล ซึ่งในขณะนั้นมีเพียงอินเทอร์เน็ตเอกซ์พลอเรอร์เท่านั้นที่มีความสามารถนี้ ต่อมาเว็บเบราว์เซอร์อื่น ๆ เช่น มอซิลลาไฟร์ฟอกซ์ ได้นำแนวคิดของ XMLHttpRequest (XHR) ไปใส่ในเบราว์เซอร์ของตนด้วย จึงเริ่มทำให้มีการใช้อย่างกว้างขวางขึ้น ซึ่งปัจจุบันกลายเป็นมาตรฐานที่ทุกเว็บเบราว์เซอร์ต้องมี

ในตอนแรกนั้นไมโครซอฟท์เป็นผู้ที่ได้นำ XMLHttpRequest (XHR) โดยใช้ใน Outlook Web Access ที่มาพร้อมกับ Microsoft Exchange Server 2000 ต่อมาเว็บไซต์อื่นเช่น กูเกิล ได้เปิดบริการใหม่ชื่อจีเมล (Gmail) ซึ่งใช้ XMLHttpRequest (XHR) เป็นหลักในการดึงข้อมูลจากเว็บเซิร์ฟเวอร์ จึงทำให้แนวคิดและเทคนิคการพัฒนาเว็บแอปพลิเคชันด้วย เอแจ็กซี เริ่มเป็นที่รู้จักกันกว้างขวางขึ้น ปัจจุบันถือว่าเป็นส่วนหลักของแนวคิดเรื่อง Web 2.0

วิธีการทำงานของเว็บแอปพลิเคชันแบบดั้งเดิมนั้น โดยปกติแล้วเมื่อผู้ใช้ทำการร้องขอข้อมูลจากเซิร์ฟเวอร์ เว็บเบราว์เซอร์จะทำการส่งข้อมูลการร้องขอโดยใช้โพรโทคอล HTTP เพื่อติดต่อกับเว็บเซิร์ฟเวอร์ และที่เว็บเซิร์ฟเวอร์จะทำการประมวลผลจากการร้องขอที่ได้รับ และส่งผลลัพธ์เป็นหน้า HTML กลับไปให้ผู้ใช้ วิธีการข้างต้นเป็นวิธีการแบบการร้องขอและการตอบรับ (Request and Response) ซึ่งผู้ใช้จะต้องรอระหว่างที่เซิร์ฟเวอร์ประมวลผลอยู่ ซึ่งเป็นหลักการทำงานแบบ Synchronous แต่การทำงานของเว็บแอปพลิเคชันที่ใช้เทคนิคเอแจ็กซ์จะเป็นการทำงานแบบ Asynchronous หรือการติดต่อสื่อสารแบบไม่ต่อเนื่อง โดยเซิร์ฟเวอร์จะทำการส่งผลลัพธ์เป็นเว็บเพจให้ผู้ใช้ทันทีโดยไม่ต้องรอให้ประมวลผลเสร็จก่อน หลังจากนั้นเว็บเพจที่ผู้ใช้ได้รับจะทำการดึงข้อมูลในส่วนต่าง ๆ ภายหลัง หรือจะดึงข้อมูลก็ต่อเมื่อผู้ใช้ต้องการเท่านั้น



รูปที่ 2.8 เปรียบเทียบการทำงานระหว่างเว็บแอปพลิเคชันแบบดั้งเดิม กับแบบที่ใช้เอแจ็กซ์



รูปที่ 2.9 เปรียบเทียบการติดต่อสื่อสารระหว่างเว็บแอปพลิเคชันแบบดั้งเดิมกับแบบที่ใช้เอเจกซ์

ข้อดีของการใช้อะซิงโครนัสคือสามารถประมวลผลได้เร็ว เรียกดูข้อมูลได้ทันที โดยไม่ต้องรีเฟรช (Refresh) หน้าเว็บที่เปิดอยู่ เทคนิคเอเจกซ์นั้นสามารถสร้าง HTML ได้ในเครื่องผู้ใช้ ทำให้ขนาดข้อมูลนั้นเล็กลงในครั้งต่อไป เพราะสามารถส่งเพียงข้อมูล และคำสั่งจาวาสคริปต์ลงมาเฉพาะส่วนที่มีการเปลี่ยนแปลง แทนการส่งข้อมูลใหม่ทั้งหมด ซึ่งทั้งนี้ขึ้นอยู่กับกรอบแบบของเว็บแอปพลิเคชันนั้นๆ

เนื่องจากการใช้เทคนิคเอเจกซ์นั้นทำให้การเปลี่ยนแปลงต่าง ๆ เช่นการแก้ไข เพิ่ม ลบ รายการข้อมูล หรือการดึงข้อมูลที่ต้องการจะค้นหานั้น สามารถทำได้ในเบื้องหลัง ทำให้ผู้ใช้รู้สึกการตอบสนองนั้น คล้ายคลึงกับโปรแกรมคอมพิวเตอร์

2.6 XMLHttpRequest [9]

XMLHttpRequest (XHR) เป็นเอพีไอที่สามารถเรียกใช้ได้จาก จาวาสคริปต์ เจสคริปต์ วีบีสคริปต์ และภาษาสคริปต์อื่นๆ ในการแลกเปลี่ยน และปรับรูปแบบ XML จากเว็บเซิร์ฟเวอร์ โดยใช้ HTTP ซึ่งสร้างการเชื่อมต่อระหว่างเว็บเบราว์เซอร์ (Client-Side) กับเว็บเซิร์ฟเวอร์ (Server-Side)

XMLHttpRequest นั้นถือเป็นส่วนสำคัญในการพัฒนาเว็บแอปพลิเคชันโดยใช้เทคนิค AJAX การเรียกใช้ XMLHttpRequest จาวาสคริปต์ตั้งแต่อินเทอร์เน็ตเอกซ์พลอเรอร์ 7 (IE7) ขึ้นไป ผู้พัฒนาสามารถเรียกใช้วัตถุ XMLHttpRequest ได้โดยตรงเหมือนเว็บเบราว์เซอร์อื่น ๆ เช่น มอซิลลา ไฟร์ฟอกซ์ โดยไม่ต้องผ่าน ActiveX ซึ่งทำให้การเรียกใช้งาน XMLHttpRequest นั้น สอดคล้องกับเว็บเบราว์เซอร์อื่น ๆ

ตัวอย่างการเรียกใช้ XMLHttpRequest ใน IE7 และเว็บเบราว์เซอร์อื่น

```
if (window.XMLHttpRequest) {
    var xmlhttp = new XMLHttpRequest ();
}
```

รูปที่ 2.10 การเรียกใช้ XMLHttpRequest ใน IE7 และเว็บเบราว์เซอร์อื่น

ตัวอย่างการเรียกใช้ XMLHttpRequest ใน IE5.x และ IE6

```
if (window.ActiveXObject) {
    var xmlhttp = new ActiveXObject ("Microsoft.XMLHTTP");
}
```

รูปที่ 2.11 การเรียกใช้ XMLHttpRequest ใน IE5.x และ IE6

เพื่อความเข้ากันได้ IE7 นั้นรองรับทั้งวิธีเดิมใน IE รุ่นก่อน และวิธีใหม่

ตัวอย่างการเรียกใช้ ทั้งสองวิธีรวมกันเพื่อรองรับ IE รุ่นเก่า รุ่นใหม่ และเว็บเบราว์เซอร์อื่นๆ

```
if (window.XMLHttpRequest) {
    var xmlhttp = new XMLHttpRequest ()
} else {
    if (window.ActiveXObject) {
        var xmlhttp = new ActiveXObject ("Microsoft.XMLHTTP");
    }
}
```

รูปที่ 2.12 การรวมการเรียกใช้ XMLHttpRequest กับทุกเว็บเบราว์เซอร์

2.7 พีเอชพี (PHP) [11]

พีเอชพี (PHP) คือ ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลิขสิทธิ์อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีใช้สำหรับจัดทำเว็บไซต์ และแสดงผลออกมาในรูปแบบ HTML

โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้ คือให้นักพัฒนาเว็บไซต์สามารถเขียน เว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว

พีเอชพีรุ่นล่าสุดคือ PHP 5.3.0 ส่วนรุ่นพัฒนาคือ PHP 6.0.0-dev

ภาษาพีเอชพี ในชื่อภาษาอังกฤษว่า PHP ซึ่งใช้เป็นคำย่อแบบกล่าวซ้ำ จากคำว่า PHP Hypertext Preprocessor หรือชื่อเดิม Personal Home Page

2.7.1 ตัวอย่างภาษาพีเอชพี

```
<?php
    echo "Hello, World!";
?>
<?
    echo "Hello World.";
?>
<script language="php">
    echo "Hello World.";
</script>
<%
    echo "Hello World.";
%>
```

รูปที่ 2.13 การใช้งานภาษาพีเอชพี

โครงสร้าง ควบคุมของ PHP จะมีความคล้ายคลึงกับ C/C++ มาก เช่น if , for , switch และมีบางส่วนที่คล้าย Perl สามารถกำหนดตัวแปรโดยไม่ต้อง นิยามก่อนได้

จุฬาลงกรณ์มหาวิทยาลัย


```

<?
  for ($i = 0; $i < 10; $i++){
    echo "Test $i";
  }
?>

```

รูปที่ 2.14 การใช้งานภาษาพีเอชพีโดยไม่กำหนดตัวแปร

2.7.2 คุณสมบัติ

การแสดงผลของพีเอชพี จะปรากฏในลักษณะHTML ซึ่งจะไม่แสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ ที่ผู้ชมเว็บไซต์สามารถอ่าน ดูและคัดลอกคำสั่งไปใช้เองได้ นอกจากนี้พีเอชพียังเป็นภาษาที่เรียนรู้และเริ่มต้นได้ไม่ยาก ความสามารถการประมวลผลหลักของพีเอชพี ได้แก่ การสร้างเนื้อหาอัตโนมัติจัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และประมวลผล การอ่านข้อมูลจากดาต้าเบส ความสามารถจัดการกับคุกกี้ ซึ่งทำงานเช่นเดียวกับโปรแกรมในลักษณะCGI คุณสมบัติอื่นเช่น การประมวลผลตามบรรทัดคำสั่ง (Command Line Scripting) ทำให้ผู้เขียนโปรแกรมสร้างสคริปต์พีเอชพี ทำงานผ่านพีเอชพี พาร์เซอร์ (PHP parser) โดยไม่ต้องผ่านเซิร์ฟเวอร์หรือเบราวเซอร์ ซึ่งมีลักษณะเหมือนกับ Cron (ใน ยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple text processing tasks ได้

บทที่ 3

การวิเคราะห์และออกแบบสถาปัตยกรรมระบบ

ระบบคลังเอกสารดิจิทัลเป็นระบบหนึ่งที่สามารถนำเสนอเอกสารผลงานวิจัย โดยการจัดเก็บเป็นแฟ้มพีดีเอฟ ทั้งนี้ในลักษณะการนำเสนอผลงานจะมีการนำข้อมูลต่างๆที่เกี่ยวกับงานวิจัยออกมาแสดงทางส่วนต่อประสานกับผู้ใช้ และท้ายสุดจะมีการเชื่อมโยงไปยังแฟ้มพีดีเอฟเพื่อดาวน์โหลดแฟ้มพีดีเอฟแล้วเปิดดูข้อมูลทั้งหมดภายในแฟ้มพีดีเอฟนั้น ๆ ซึ่งในงานวิจัยนี้ได้มีแนวคิดในการแสดงข้อมูลภายในแฟ้มพีดีเอฟโดยใช้รูปภาพแสดงแทน ผู้วิจัยจึงได้ทำการวิเคราะห์และออกแบบระบบ โดยศึกษาจากการค้นหาและรวบรวมเครื่องมือ วิธีการ และปัญหาในการแปลงแฟ้มเอกสารจากพีดีเอฟเป็นแฟ้มรูปภาพและนำไปแสดงผ่านส่วนต่อประสานกับผู้ใช้ ซึ่งมีรายละเอียดดังต่อไปนี้

3.1 การวิเคราะห์ความต้องการ

3.1.1 การแปลงแฟ้มพีดีเอฟเป็นแฟ้มรูปภาพ

ในงานวิจัยนี้จำเป็นที่จะต้องมีการนำแฟ้มพีดีเอฟที่มีอยู่ในระบบคลังเอกสารดิจิทัล หรือบนเครื่องแม่ข่ายมาแปลงเป็นแฟ้มรูปภาพเพื่อที่จะสามารถนำไปแสดงต่อผู้ใช้ ซึ่งในการวิเคราะห์ชนิดของแฟ้มรูปภาพจากความต้องการที่จะแสดงผลให้ได้เหมือนแฟ้มพีดีเอฟที่เป็นต้นฉบับจึงได้เลือกแฟ้มเจเพ็ก (JPEG) ซึ่งเป็นที่นิยมในการแสดงผลรูปภาพบนเว็บไซต์เพราะแฟ้มชนิดนี้สามารถแสดงผลของสีได้สมจริงและมีขนาดไฟล์เล็ก

3.1.2 การนำแฟ้มรูปภาพมาแสดงในรูปแบบเว็บแอปพลิเคชัน

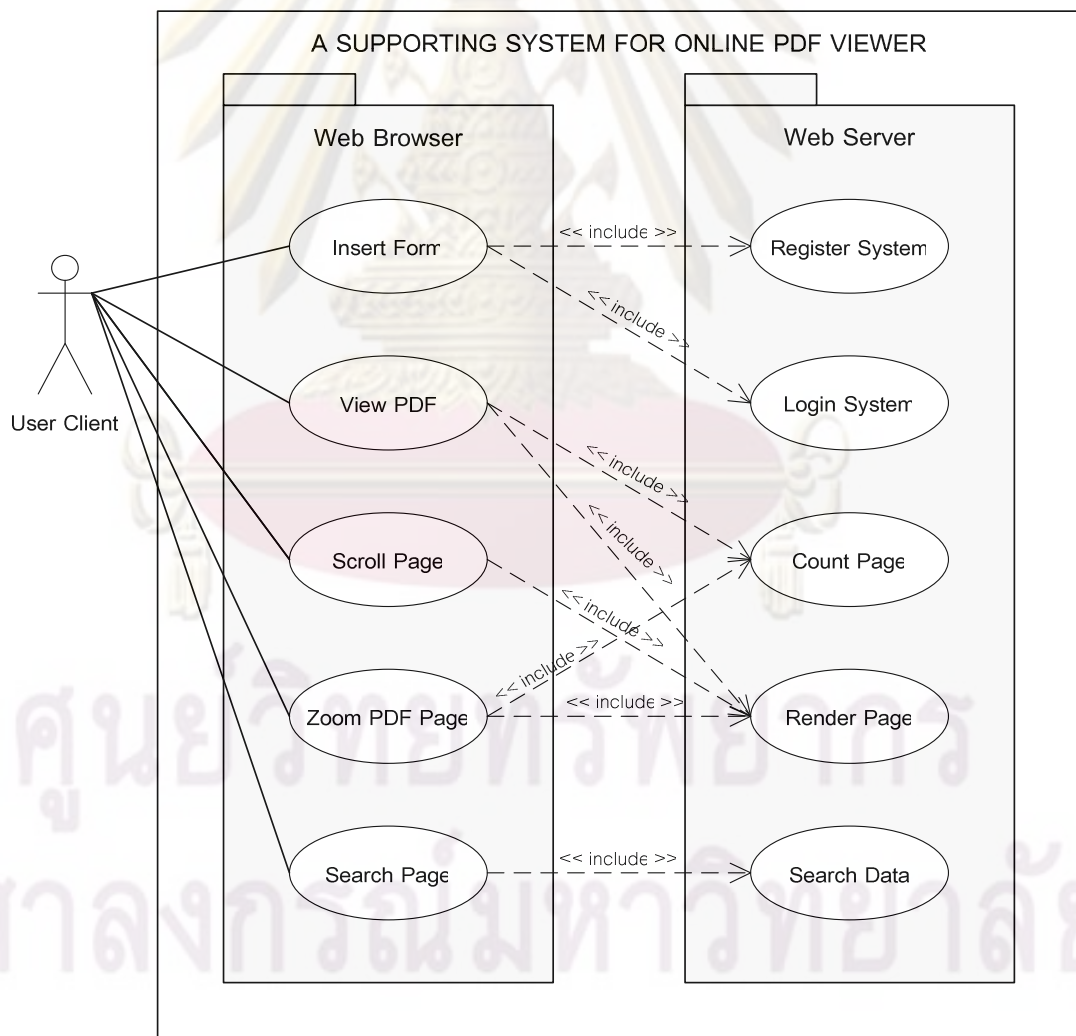
จุดประสงค์ของงานวิจัยนี้ต้องการที่จะนำเสนอข้อมูลที่ได้จากการแปลงแฟ้มพีดีเอฟเป็นแฟ้มเจเพ็ก โดยในการนำเสนอนั้นจะเป็นลักษณะรูปแบบของการทำงานที่มีความคล้ายกับโปรแกรมแอปพลิเคชันสำหรับเปิดแฟ้มพีดีเอฟ ซึ่งจากการค้นหารูปแบบการแสดงผลที่สามารถทำได้ในลักษณะเช่นนี้ จำเป็นที่จะต้องนำความรู้ในเรื่องของการเขียนเว็บ และจาวาสคริปต์ ซึ่งต้องใช้เทคนิคในการเขียนหลายอย่างมาประกอบให้ทำงานร่วมกัน

3.1.3 การค้นหาข้อมูลภายในแฟ้มพีดีเอฟ

การแสดงผลเนื้อหาภายในแฟ้มพีดีเอฟ เพื่อความสะดวกในการดูข้อมูลจำเป็นที่จะต้องมีการค้นหาข้อมูลต่าง เช่น คำที่ผู้ใช้ต้องการหรือหัวข้อหลักภายในแฟ้มพีดีเอฟ และเพื่อให้ครอบคลุมขอบเขตของงานวิจัยนี้ ฉะนั้นในการทำงานของส่วนนี้จะต้องเป็นการทำงานในฝั่งเครื่องแม่ข่าย เพื่อทำหน้าที่ในการเข้าไปค้นหาข้อมูลที่ต้องการภายในแฟ้มพีดีเอฟต้นฉบับที่เก็บไว้บนเครื่องแม่ข่าย และนำข้อมูลที่ได้ส่งกลับมาแสดงให้กับผู้ใช้

3.2 การวิเคราะห์ระบบ

ผู้วิจัยได้ทำการวิเคราะห์ระบบ โดยใช้แผนภาพยูสเคสดังแสดงในรูปที่ 3.1 โดยมีคำอธิบายยูสเคสแสดงในภาคผนวก ก.

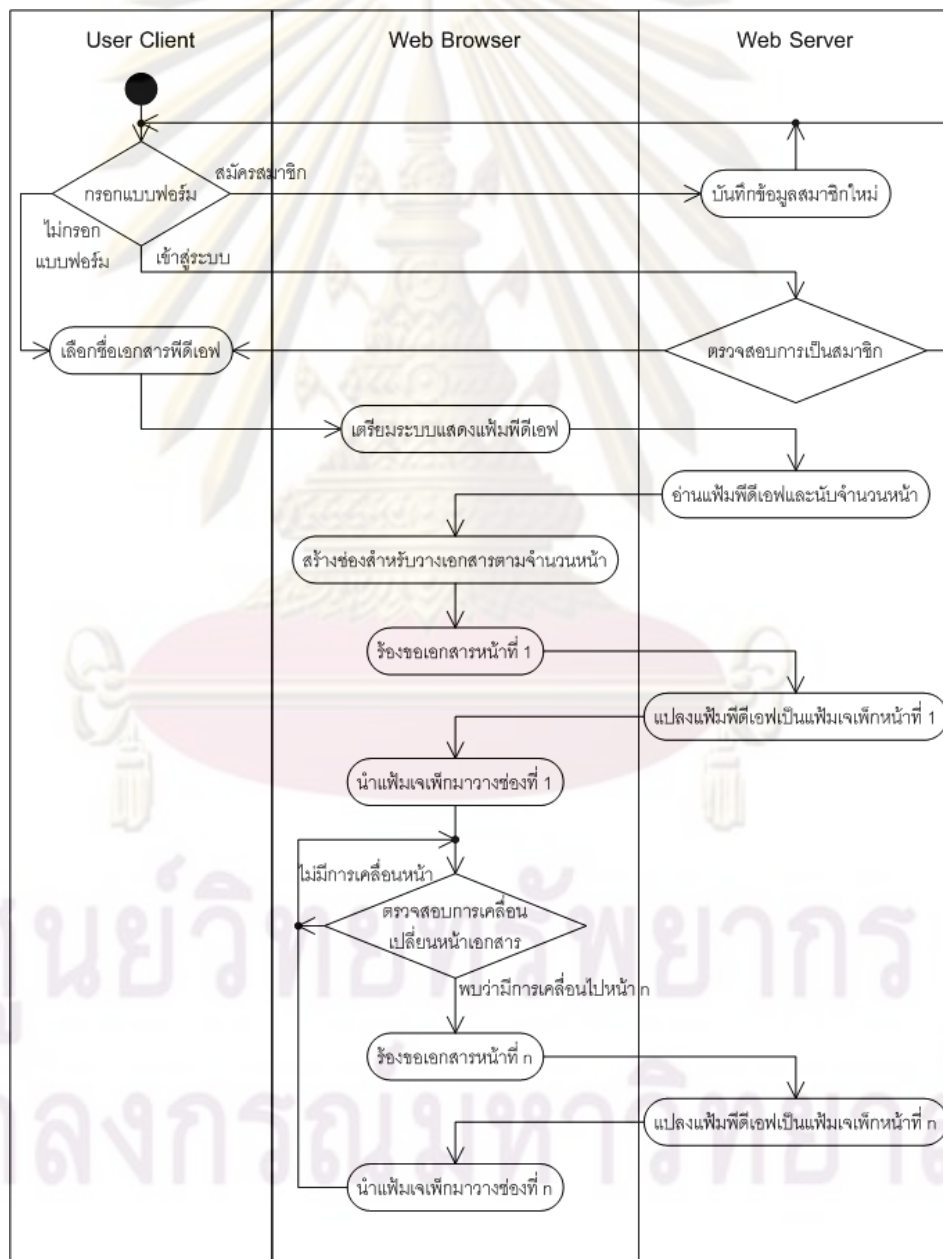


รูปที่ 3.1 แผนภาพยูสเคสการวิเคราะห์ระบบ

3.3 การออกแบบระบบ

3.3.1 การออกแบบขั้นตอนการทำงานของระบบ

จากความสัมพันธ์ของการทำงานระหว่างผู้ใช้กับเครื่องแม่ข่าย โดยมีเว็บเบราว์เซอร์เป็นตัวกลางในการทำงาน ทั้งนี้ในการทำงานร่วมกันของระบบทั้งหมด จะต้องมีลำดับขั้นตอนการทำงานโดยมีผู้ใช้เป็นผู้เริ่มกระบวนการ และใช้งานบนระบบสนับสนุนจนถึงสิ้นสุดการทำงาน ซึ่งผู้วิจัยได้นำเสนอขั้นตอนด้วยแผนภาพแสดงขั้นตอนการทำงาน (Activity Diagram) ดังรูปที่ 3.2



รูปที่ 3.2 การออกแบบขั้นตอนการทำงานของระบบ

3.3.1.1 การกรอกแบบฟอร์มเพื่อลงทะเบียนและเข้าสู่ระบบ

สำหรับการใช้งานระบบแสดงผลแฟ้มพีดีเอฟ ลำดับแรกจะมีส่วนของการรับข้อมูลจากผู้ใช้เพื่อสมัครสมาชิก และเข้าสู่ระบบ เพื่อที่จะสามารถใช้งานระบบแสดงผลแฟ้มพีดีเอฟได้อย่างถูกต้อง โดยผู้ใช้จำเป็นที่จะต้องสมัครสมาชิกก่อนจึงจะสามารถเข้าสู่ระบบได้ ทั้งนี้ผู้ใช้ที่เข้ามาใช้งานทั้งที่เป็นหรือไม่ได้เป็นสมาชิก สามารถผ่านขั้นตอนนี้ได้

ในการที่ผู้ใช้ที่เป็นสมาชิกและทำการเข้าสู่ระบบ ทางฝั่งเครื่องแม่ข่ายจะทำการแจ้งการเข้าสู่ระบบของผู้ใช้ไว้ในตัวแปร SESSION ที่อยู่บนระบบเครื่องแม่ข่ายซึ่งจะใช้ประโยชน์ในการอนุญาตการเข้าถึงข้อมูลแฟ้มพีดีเอฟ

3.3.1.2 การเตรียมระบบแสดงผลแฟ้มพีดีเอฟ

ในส่วนของผู้ใช้ที่อยู่บนเครื่องผู้ใช้ ต้องการเปิดแฟ้มพีดีเอฟระบบจะทำงานอยู่บนเว็บเบราว์เซอร์ โดยที่เบราว์เซอร์จะทำการเรียกแฟ้มโปรแกรมภาษาจาวาสคริปต์ที่อยู่บนฝั่งเครื่องแม่ข่ายมาทำการแปลโปรแกรม (Compile) เพื่อสร้างและเตรียมระบบสนับสนุนพื้นฐาน โครงสร้างลักษณะของส่วนต่อประสานกับผู้ใช้ ซึ่งในส่วนนี้จะมีการสร้างขึ้นก่อนทุกครั้งที่มีการเรียกใช้งาน โดยมีรูปแบบการเรียกใช้จาวาสคริปต์ดังรูปที่ 3.3

```
<script type="text/javascript" src="incpdf/js/pdfbook.js"></script>
<script>
function init() {
    myBook1 = new cpcu.pdfbooks.PDF($cpcu("myBook"));
}
</script>
```

รูปที่ 3.3 การเรียกใช้งานจาวาสคริปต์

สำหรับตัวอย่างข้างต้นไปการเรียกใช้งานจาวาสคริปต์ ซึ่งสามารถนำคำสั่งนี้ไปวางบนหน้าเว็บที่มีคำสั่งภาษา HTML ได้ และตำแหน่งของแฟ้มจาวาสคริปต์ “incpdf/js/pdfbook.js” จะกล่าวถึงในบทต่อไป ส่วน “myBook” จะเป็นชื่อของวัตถุที่อยู่บนหน้าเว็บที่ต้องการให้ระบบแสดงผลแฟ้มพีดีเอฟไปแสดงอยู่ภายในวัตถุนั้น

3.3.1.3 การอ่านแฟ้มพีดีเอฟและนับจำนวนหน้า

เมื่อเว็บเบราว์เซอร์ของผู้ใช้เตรียมระบบเสร็จสิ้น ระบบจะใช้เอแจ็กต์ส่งคำสั่งขอจำนวนหน้าทั้งหมดของแฟ้มพีดีเอฟไปยังเครื่องแม่ข่าย จากนั้นเครื่องแม่ข่ายจะทำการนับจำนวนหน้าของแฟ้มพีดีเอฟที่ต้องการ ในส่วนการทำงานนี้จะใช้โปรแกรมภาษา PHP ซึ่งจะใช้เทคนิคในการนับจำนวนหน้าแฟ้มพีดีเอฟดังตัวอย่างของฟังก์ชัน ดังรูปที่ 3.4

```
var httpValue = Util.createRequestObject();
httpValue.open('get', phpPath + 'incpdf/php/pdfbook.php?a=count&pdfid=' + this.pdfid,
    false);
httpValue.send(null);
this.totalPage = httpValue.responseText;
this.pageStatus.innerHTML = "Page " + this.pageCurrent + " / " + this.totalPage;
```

รูปที่ 3.4 การใช้เอแจ็กต์เพื่อขอจำนวนหน้าพีดีเอฟ

```
<?php
function count_pages($pdfname) {
    $pdftext = file_get_contents($pdfname);
    $num = preg_match_all("/PageW/", $pdftext, $dummy);
    return $num;
}
?>
```

รูปที่ 3.5 ฟังก์ชันการนับจำนวนหน้าในแฟ้มพีดีเอฟ

จากการศึกษาพบว่าแฟ้มพีดีเอฟจะมีการระบุรูปแบบของคำสั่งในการค้นหาของเอกสาร ซึ่งเมื่อได้ทำการอ่านข้อมูลดิบของแฟ้มพีดีเอฟแล้ว จะพบว่ามีคำสั่งที่มีรูปแบบคือ /PageXXX (XXX หมายถึงข้อมูลตัวอักษรใด ๆ ไม่เจาะจง) โดยจะทำการนับคำสั่งที่มีรูปแบบเช่นนี้ว่ามีจำนวนทั้งหมดเท่าใด นั่นก็คือจำนวนหน้าของแฟ้มพีดีเอฟ และเมื่อได้จำนวนหน้าแล้วระบบบนเครื่องแม่ข่ายก็จะทำการส่งค่าที่เป็นตัวอักษรหรือตัวเลข ซึ่งใน ณ ที่นี้คือตัวเลขจำนวนหน้ากลับมายังเว็บเบราว์เซอร์ของผู้ใช้ที่ทำการร้องขอเพื่อดำเนินการทำงานต่อไป แต่ก่อนที่ระบบจะส่งข้อมูลจำนวนหน้าไปนั้น จะมีการตรวจสอบ SESSION ของผู้ใช่ว่ามีการเข้าสู่ระบบข้างต้นก่อน

หรือไม่ ซึ่งผลที่ได้คือในกรณีที่ไม่ได้เข้าสู่ระบบ ผู้ใช้จะสามารถดูข้อมูลได้เพียง 5 หน้าแรกเท่านั้น ระบบก็จะส่งข้อมูลจำนวน 5 กลับไปแทนจำนวนหน้าทั้งหมด

3.3.1.4 การสร้างช่องสำหรับวางเอกสารตามจำนวนหน้า

ในการจัดเรียงเอกสารในแต่ละหน้าก่อนที่จะมีการนำรูปมาวางนั้น จะต้องเตรียมพื้นที่ที่ถูกต้องสำหรับวางเอกสารในแต่ละหน้า ดังนั้นเมื่อได้รับจำนวนหน้าของแฟ้มพีดีเอฟแล้ว ก็จะมีการสร้างช่องสี่เหลี่ยมตามจำนวนที่ได้รับ โดยการเรียงกันลงมาตามแนวดิ่ง ซึ่งแต่ละช่องจะมีการเว้นระยะห่างและมีการบอกเลขหน้านั้น ๆ

3.3.1.5 การร้องขอเอกสารหน้าที่ 1

หลังจากระบบได้ทำการเตรียมช่องสำหรับวางเอกสารแล้วจะทำการร้องขอข้อมูลเอกสารหน้าที่ 1 ไปยังเครื่องแม่ข่ายโดยตรง ซึ่งเครื่องแม่ข่ายจะทำการแปลงแฟ้มพีดีเอฟเฉพาะหน้าที่ 1 เป็นแฟ้มเจเพ็กโดยไม่ได้ทำการจัดเก็บแฟ้มเจเพ็กที่ได้ไว้บนเครื่องแม่ข่าย แต่จะทำการส่งข้อมูลแฟ้มไปยังเครื่องของผู้ใช้เพื่อไปแสดงผลเว็บเบราว์เซอร์ทันที สำหรับการแปลงข้อมูลนั้นจะใช้คำสั่งของโปรแกรม ImageMagick ที่ทำงานอยู่บนเครื่องแม่ข่ายซึ่งมีรูปแบบการใช้งานดังรูปที่ 3.6

```
$im = new Imagick();
$im -> setResolution($resolution, $resolution);
$im -> readImage("${pdf}[{$page}]");
$im -> scaleImage($widthSize,0);
$im -> setImageFormat('jpeg');
header('Content-Type: image/jpeg');
echo $im;
```

รูปที่ 3.6 คำสั่งแปลงข้อมูลแฟ้มพีดีเอฟเป็นแฟ้มเจเพ็ก

โดยมีตัวแปร \$resolution หมายถึงความละเอียดของภาพหน้าเอกสารที่จะสร้างขึ้น มีหน่วยความละเอียดเป็น Pixel ตอนนี้ทั้งด้านกว้างและสูง ส่วนตัวแปร \$pdf จะหมายถึงตำแหน่งและชื่อของแฟ้มพีดีเอฟที่ต้องการเปิด โดยที่ตัวแปร \$page เป็นค่าสำหรับบอก

เลขที่ของหน้าในแฟ้มพีดีเอฟที่ต้องการแปลงข้อมูล ซึ่งค่านี้จะรับมาจากการร้องขอของเอเจ็ทส์ว่า ต้องการขอเอกสารหน้าใด

3.3.1.6 การนำแฟ้มเจพีคมาวางในช่องเอกสาร

หลังจากการที่ได้มีการร้องขอเอกสารในหน้าที่ต้องการแล้วนั้น ระบบจะได้รับข้อมูลแฟ้มเจพีคกลับมาที่เว็บเบราว์เซอร์ ซึ่งจะมีคำสั่งสำหรับรองรับข้อมูลที่ได้นำเป็นแสดงลงในช่องของหน้าที่ทำการร้องขอเพื่อแสดงผลให้กับผู้ใช้

3.3.1.7 การตรวจสอบการเคลื่อนเปลี่ยนหน้าเอกสาร

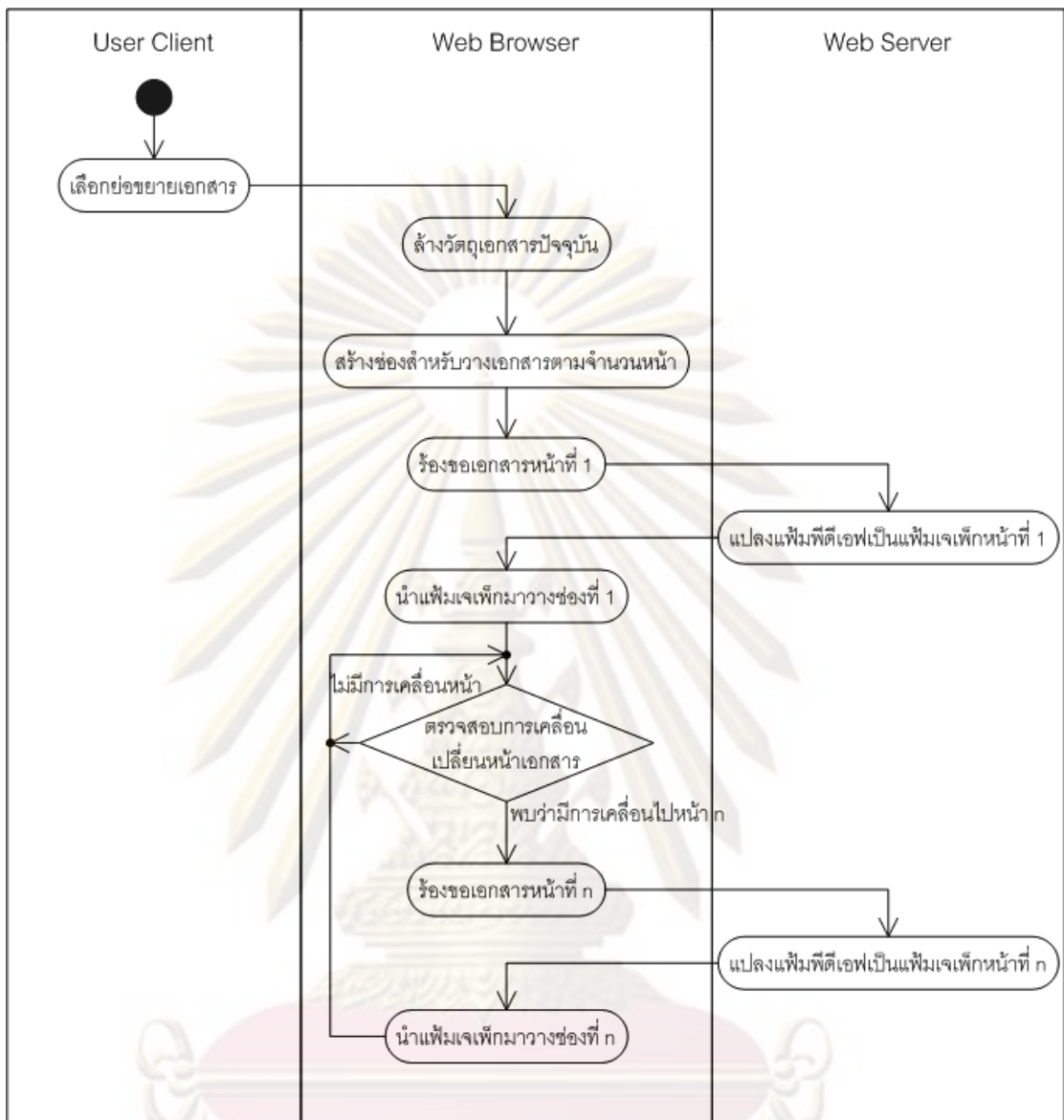
การร้องขอเอกสารหน้าที่ 1 ที่ได้กล่าวมาข้างต้นนั้น เป็นขั้นตอนการทำงานหลักซึ่งจะต้องทำทุกครั้งที่มีการใช้งานระบบ แต่หลังจากที่ได้รับเอกสารหน้าที่ 1 และทำการนำมาวางในตำแหน่งที่ถูกต้องแล้ว หลังจากนั้นระบบจะมีคำสั่งที่ใช้สำหรับตรวจสอบการใช้งานของผู้ใช้ว่ามีการเคลื่อนหน้าที่กำลังอ่านไปยังตำแหน่งหน้าอื่นหรือไม่ เมื่อพบว่ามี การเคลื่อนเปลี่ยนไปยังหน้าอื่นก็จะทำการร้องขอเอกสารหน้านั้น ๆ โดยมีขั้นตอนการทำงานเช่นเดียวกับข้อที่ 3.3.2.5 และ 3.3.2.6 ซึ่งในส่วนนี้ระบบจะสั่งให้มีการตรวจทุก ๆ 1 วินาที เนื่องจากเพื่อไม่ให้มีการทำงานของระบบมากเกินไปและเป็นการเว้นระยะเวลาสำหรับการเคลื่อนไปยังหน้าอื่นแล้วไม่จำเป็นต้องร้องขอเอกสารมาทุกหน้าที่ทำการเคลื่อนผ่าน

สำหรับหลักการตรวจสอบสถานะของตำแหน่งหน้าที่แสดงอยู่นั้น ได้มีการนำค่าของระยะของการเคลื่อนที่มาเทียบกับระยะของความสูงเอกสารที่เรียงกันทั้งหมด แล้วคิดเป็นตำแหน่งของหน้าเอกสารนั้น โดยได้สูตรดังนี้

$$\text{ตำแหน่งหน้าเอกสารปัจจุบัน} = \frac{(\text{จำนวนหน้าทั้งหมด} \times \text{ความสูงของการเคลื่อนหน้าเอกสาร})}{\text{ผลรวมของความสูงเอกสารทั้งหมด}}$$

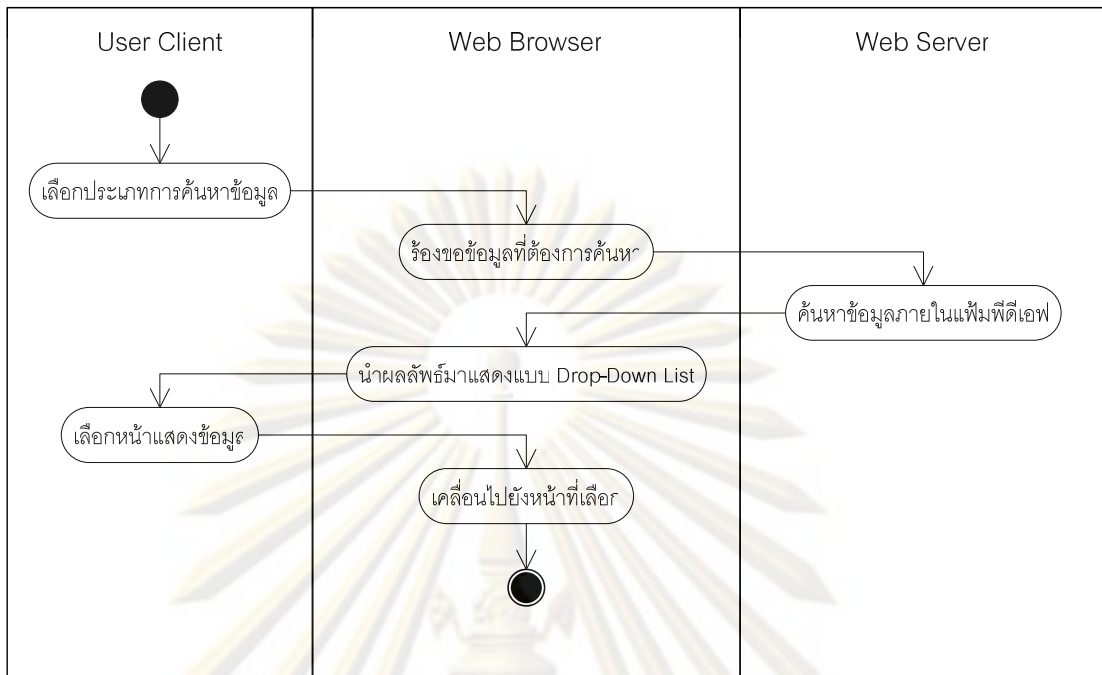
3.3.2 การออกแบบการทำงานของฟังก์ชัน

การใช้งานไม่เพียงแต่การดูข้อมูลเอกสารในลักษณะเดียว แต่ระบบนั้นสามารถทำการย่อขยายขนาดของเอกสารที่ผู้ใช้เปิด อีกทั้งยังสามารถค้นหาข้อมูลที่เป็นหัวข้อหลัก และข้อมูลส่วนที่เป็นรูปภาพหรือตาราง ไปจนถึงค่าที่ผู้ใช้ต้องการค้นหาภายในเอกสาร ผู้วิจัยได้ออกแบบฟังก์ชันการย่อขยาย และระบบการค้นหาข้อมูล ด้วยแผนภาพแสดงขั้นตอนการทำงาน (Activity Diagram) ดังรูปที่ 3.7 และรูปที่ 3.8 ตามลำดับ



รูปที่ 3.7 การออกแบบขั้นตอนการย่อขยายเอกสารระบบแสดงผลแฟ้มพีดีเอฟ

จากรูปได้แสดงถึงการที่ผู้ใช้งานต้องการย่อขยายเอกสาร ซึ่งได้มีการกำหนดให้มีระดับขนาดการแสดงผลเอกสารทั้งหมด 10 ระดับจาก 10% ถึง 800% โดยมีค่าเริ่มต้นอยู่ที่ 100% และเมื่อมีการเลือกการย่อขยายในระดับต่าง ๆ จะมีการล้างข้อมูลวัตถุในส่วนที่เป็นข้อมูลเอกสารที่แสดงออกทั้งหมด ซึ่งในขั้นตอนต่อไปจะเป็นการสร้างข้อมูลเอกสารใหม่ โดยมีขนาดของเอกสารที่กำหนดในตัวแปร $widthSize$ ตามที่ได้ทำการเลือกไป ในส่วนของการสร้างเอกสารใหม่นี้จะมีขั้นตอนการทำงานเหมือนกับข้อที่ 3.3.2.4 เป็นต้นไป และการกำหนดขนาดของเอกสารที่นำมาแสดงในตัวแปร $widthSize$ นั้นจะมีขนาดความกว้างโดยเทียบจากขนาด 100% มีค่าเท่ากับ 715 จุด (Pixel) สำหรับความสูงจากขึ้นอยู่กับอัตราส่วนของแฟ้มพีดีเอฟ



รูปที่ 3.8 การออกแบบขั้นตอนการค้นหาข้อมูลภายในแฟ้มพีดีเอฟ

สำหรับการค้นหาข้อมูลภายในแฟ้มพีดีเอฟจะขออธิบายเป็นลำดับได้ดังนี้

- 1) เลือกประเภทการค้นหาข้อมูล ในการค้นหาผู้วิจัยได้กำหนดประเภทของการค้นหาข้อมูลออกเป็น 2 ประเภทคือ ค้นหาคำ และค้นหาข้อมูลตามหัวข้อหลักในเอกสาร ซึ่งผู้ใช้งานจะสามารถเลือกให้แสดงข้อมูลประเภทใด
- 2) เมื่อผู้ใช้แสดงความต้องการในการค้นหาข้อมูล ในส่วนของจาวาสคริปต์จะใช้เอแจ็กส์ส่งคำสั่งการร้องขอข้อมูลไปยังเครื่องแม่ข่าย ซึ่งได้มีการแนบข้อมูลความต้องการประเภทข้อมูลไปด้วย
- 3) เครื่องแม่ข่ายจะทำหน้าที่ในการค้นหาข้อมูลที่ได้มีการร้องขอมา โดยได้แบ่งวิธีการค้นหาข้อมูลออกเป็น 2 ระบบตามประเภทของการค้นหาข้อมูลดังต่อไปนี้
 - การค้นหาคำ จะให้วิธีการแปลงข้อมูลแฟ้มพีดีเอฟเป็นข้อมูลตัวอักษรซึ่งจะอยู่ในตัวแปร \$text ดังรูปที่ 3.9 และทำการค้นหาคำภายในข้อมูลตัวอักษรที่ได้ โดยค้นหาคำที่ละหน้า จากนั้นจะระบุเป็นรายการตำแหน่งหน้าของเอกสาร

```

$text = shell_exec('pdftotext -enc UTF-8 '.$pdf.' -');
$text = explode(chr(12), $text);
for ($p = 0; $p < count($text) - 1; $p++) {
    if ($p >= $_SESSION['bookpdftotalpage'].$fileName]) {
        break;
    }
    $datapage = explode($keyword, $text[$p]);
    for ($k = 1; $k < count($datapage); $k++) {
        $before = str_replace("\n", " ", $datapage[$k - 1]);
        $before = explode(' ', $before);
        $before = $before[count($before) - 1];
        $content = str_replace("\n", "", $datapage[$k]);
        $content = explode(' ', $content);
        echo ($p + 1).'=='.$before.$keyword.$content[0].'|';
    }
}

```

รูปที่ 3.9 คำสั่งการค้นหาคำภายในแฟ้มพีดีเอฟ

- การค้นหาหัวข้อหลักภายในเอกสาร จะให้วิธีการแปลงข้อมูลแฟ้มพีดีเอฟเป็นข้อมูลตัวอักษรซึ่งจะอยู่ในตัวแปร \$text เช่นเดียวกันดังรูปที่ 3.10 ค้นหาทีละหน้า และตรวจสอบหาข้อมูลที่ประกอบไปด้วยตัวเลข จุด และข้อความ ซึ่งคำเหล่านี้มักจะเป็นหัวข้อ ชื่อรูป ชื่อตาราง โดยใช้ Regular Expression [13] โดยมีรูปแบบเป็น $/(W[0-9])+(\.){1}(.)+/$ จากนั้นจะระบุเป็นรายการตำแหน่งหน้าของเอกสาร

```

$text = shell_exec('pdftotext -enc UTF-8 '.$pdf.' -');
$text = str_replace('.', '', $text);
$text = explode(chr(12), $text);
for ($p = 0; $p < count($text) - 1; $p++) {
    if ($p >= $_SESSION['bookpdftotalpage'].$fileName]) {
        break;
    }
    $datapage = explode("\n", $text[$p]);
    for ($i = 0; $i < count($datapage); $i++) {
        if (preg_match('/^(\\W|[0-9])+\\.\\{1\\}\\(\\.\\)+/', $datapage[$i])) {
            $tmp = explode(' ', $datapage[$i]);
            $tmpCount = count($tmp);
            if ($tmpCount > 3) $tmpCount = 3;
            $temp = "";
            for ($j = 0; $j < $tmpCount; $j++) {
                $temp .= $tmp[$j];
            }
            echo ($p + 1).'=='.$temp.'|||';
        }
    }
}
}

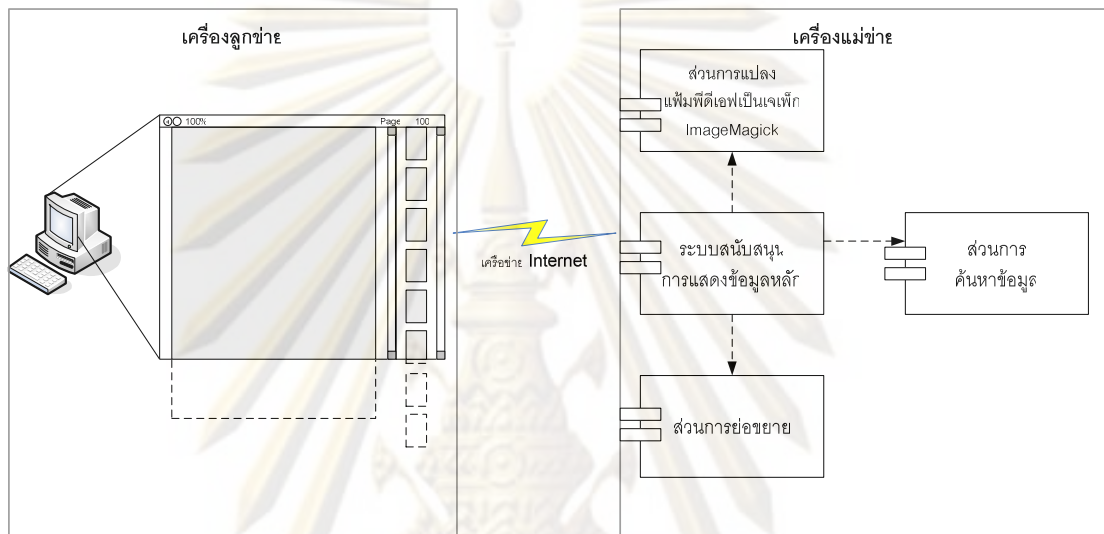
```

รูปที่ 3.10 คำสั่งการค้นหาหัวข้อหลักภายในแฟ้มพีดีเอฟ

- 4) หลังจากรับข้อมูลที่ได้จากเครื่องแม่ข่ายแล้ว จาวาสคริปต์จะทำการส่งข้อมูลที่เป็นรายการเข้าไปยัง Drop-Down List ซึ่งอยู่บนหน้าระบบแสดงผลแฟ้มพีดีเอฟ เพื่อสำหรับให้ผู้ใช้เลือกความต้องการเคลื่อนหน้าเอกสารไปยังยั้งนั้น ๆ
- 5) ผู้ใช้ทำการเลือกหน้าที่มีข้อมูลที่ต้องการค้นหาอยู่จาก Drop-Down List
- 6) หน้าเอกสารจะทำการเคลื่อนหน้าไปยังตำแหน่งหน้าที่ผู้ใช้ได้เลือกไว้

3.3.3 สถาปัตยกรรมของระบบ

จากการออกแบบระบบตามที่ได้นำเสนอไปแล้วนั้น ผู้วิจัยได้ทำการพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจเพ็ทเพื่อให้สามารถใช้งานดูเอกสารได้จริงโดยใช้เทคนิคการแปลงข้อมูลจากแฟ้มพีดีเอฟเป็นแฟ้มรูปภาพเจเพ็ทแทน ซึ่งสามารถสรุปเป็นสถาปัตยกรรมของระบบได้ดังรูปที่ 3.11



รูปที่ 3.11 แผนภาพสถาปัตยกรรมของระบบ

บทที่ 4

การพัฒนาระบบ

4.1 ขั้นตอนการพัฒนาระบบ

4.1.1 ซอฟต์แวร์ที่ใช้ในการพัฒนาในเครื่องแม่ข่าย

- 1) ระบบปฏิบัติการ Ubuntu รุ่น 9.04
- 2) ระบบ Apache Server
- 3) ระบบ PHP5
- 4) โปรแกรม ImageMagick
- 5) โปรแกรม XPDF

4.1.2 ซอฟต์แวร์ที่ใช้ในเครื่องผู้ใช้

โปรแกรมเว็บเบราว์เซอร์อย่างน้อย 1 โปรแกรม เช่น Internet Explorer, Mozilla Firefox, Apple Safari, Opera เป็นต้น

4.1.3 การพัฒนาระบบ

ในการพัฒนาระบบ ผู้วิจัยจะแยกการพัฒนาออกเป็น 2 ส่วน คือ

4.1.3.1 การพัฒนาในส่วนของการทำงานบนฝั่งเครื่องแม่ข่าย

จากการวิเคราะห์ออกแบบระบบมาแล้วนั้น ผู้วิจัยได้ทำการพัฒนาให้มีโปรแกรมบนฝั่งเครื่องแม่ข่ายที่สามารถทำงานได้หลายหน้าที่ ซึ่งได้ใช้โปรแกรมภาษา PHP ในการพัฒนา โดยได้อธิบายขั้นตอนและวิธีการในบทที่ 3 ในส่วนของฝั่งเครื่องแม่ข่าย และสามารถสรุปหน้าที่การทำงานได้ คือ นับจำนวนหน้าของแฟ้มพีดีเอฟ, แปลงแฟ้มพีดีเอฟเป็นแฟ้มเจพีค และค้นหาข้อมูลภายในแฟ้มพีดีเอฟ

4.1.3.2 การพัฒนาในส่วนของการทำงานบนฝั่งเครื่องผู้ใช้

ในส่วนนี้เป็นส่วนสำคัญที่จำเป็นที่จะต้องใช้เทคนิคในการเขียนฟังก์ชันให้ทำงานบนเบราว์เซอร์ได้หลายโปรแกรมตามขอบเขตของงานวิจัยนี้โดยใช้วิธีการในการทำให้ลักษณะการใช้งานของระบบใช้ได้เหมือนกันการใช้โปรแกรมประยุกต์สำหรับเปิดแฟ้มพีดีเอฟ ใน

การพัฒนากระบวนการนี้จะใช้จาวาสคริปต์สำหรับพัฒนาโปรแกรมซึ่งทำงานอยู่บนเว็บเบราว์เซอร์ของผู้ใช้ โดยมีขั้นตอนการทำงานตามที่ได้กล่าวมาในบทที่ 3

4.2 ตัวอย่างส่วนต่อประสานผู้ใช้ (User Interface)

สำหรับส่วนต่อประสานผู้ใช้ เป็นการพัฒนาหน้าจอที่ทำหน้าที่เชื่อมต่อระหว่างผู้ใช้ และระบบสนับสนุน โดยออกแบบตามขั้นตอนการทำงานของระบบ ทั้งนี้ได้ทำการออกแบบเพื่อทดสอบใช้ร่วมกับคลังปัญญาจุฬา ฯ เพื่อประเทศไทย

4.2.1 หน้าลงทะเบียนและเข้าสู่ระบบ

เริ่มแรกของการใช้งานระบบจะแบ่งเป็น 2 ส่วนคือส่วนที่เป็นแบบฟอร์มสำหรับให้ผู้ใช้กรอกข้อมูลสำหรับสมัครสมาชิกและเข้าสู่ระบบ อีกส่วนที่เป็นรายการชื่อของแฟ้มพีดีเอฟดังรูปที่ 4.1

DSpace™ Chulalongkorn University Institutional Repository

Login

Email:

Password:

Login Register

การดำเนินงานตามสัญญาจ้างก่อสร้างของงานรัฐในประเทศไทย และเวียดนาม
การออกแบบและพัฒนาระบบสารสนเทศปริภูมิผ่านเว็บเซอวิส
การค้นหาเว็บเซอวิสเชิงความหมายด้วยข้อกำหนดอวาล์-เอส โพรเซสโมเดล
การพัฒนาส่วนต่อประสานของโปรแกรมดีสเปซสำหรับสถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย
การพัฒนาอุปกรณ์วัดค่าฮาร์ดแวร์และเซอร์วิสเชิงอิเล็กทรอนิกส์สำหรับการฉายรังสีครั้งเดียว

รูปที่ 4.1 ส่วนต่อประสานผู้ใช้ของระบบลงทะเบียนและเข้าสู่ระบบ

4.2.2 หน้าแสดงข้อมูลเบื้องต้นและรายละเอียดของวิทยานิพนธ์

ส่วนนี้จะนำหน้าที่มีอยู่แล้วในระบบของคลังปัญญาจุฬา ฯ เพื่อประเทศไทยมาใช้ โดยจะมีการเปลี่ยนจุดเชื่อมโยงให้ไปยังระบบแสดงผลแฟ้มพีดีเอฟที่ได้ทำการพัฒนาดังรูปที่ 4.2

effective project planning and having appropriate risk assurances.

Description: วิทยานิพนธ์ (วศ.ม.)--จุฬาลงกรณ์มหาวิทยาลัย, 2549
Degree Name: วิศวกรรมศาสตรมหาบัณฑิต
Degree Level: ปริญญาโท
URI: http://hdl.handle.net/123456789/11903
ISBN: 9741434073
Appears in Collections: [Theses](#)

Files in This Item:

File	Description	Size	Format	
piya.pdf		1812Kb	Adobe PDF	View/Open

[Show full item record](#)

All items in DSpace are protected by copyright, with all rights reserved.

W3C XHTML 1.0

DSpace Software Copyright © 2002-2005 MIT and Hewlett-Packard - [Feedback](#)

รูปที่ 4.2 รายละเอียดวิทยานิพนธ์ตำแหน่งที่มีการเปลี่ยนจุดเชื่อมโยง

4.2.3 ระบบแสดงผลแฟ้มพีดีเอฟ

เป็นส่วนที่ได้พัฒนาขึ้นสำหรับเป็นหน้าหลักในการเปิดแฟ้มพีดีเอฟ

การค้นคืนงานตามสัญญาที่ก่อตั้งระหว่างหน่วยงานรัฐในประทศลาว ไทย และเวียดนาม

ศูนย์วิทยุทรัพยากร

นาง ปิยะ เจริญเจริญ

Page 1 / 145

Pub 1

Pub 2

รูปที่ 4.3 ส่วนต่อประสานผู้ใช้ระบบแสดงผลแฟ้มพีดีเอฟ

4.3 การจัดเก็บแฟ้มของโปรแกรมที่พัฒนา

จากการพัฒนาระบบข้างต้นจะเน้นเฉพาะส่วนของแฟ้มที่เป็นการพัฒนาระบบแสดงผลแฟ้มพีดีเอฟเท่านั้น ซึ่งได้มีการแยกออกเป็น 2 ส่วน ดังนี้คือ

4.3.1 เพิ่มโปรแกรมที่ทำงานอยู่บนฝั่งเครื่องแม่ข่าย

พัฒนาโปรแกรมบนเพิ่มที่มีชื่อว่า “pdfbook.php” มีหน้าที่ในการจัดการเกี่ยวกับระบบที่ทำงานอยู่บนฝั่งเครื่องแม่ข่าย เป็นโปรแกรมภาษาพีเอชพี (PHP)

4.3.2 เพิ่มโปรแกรมที่ทำงานอยู่บนฝั่งเครื่องผู้ใช้

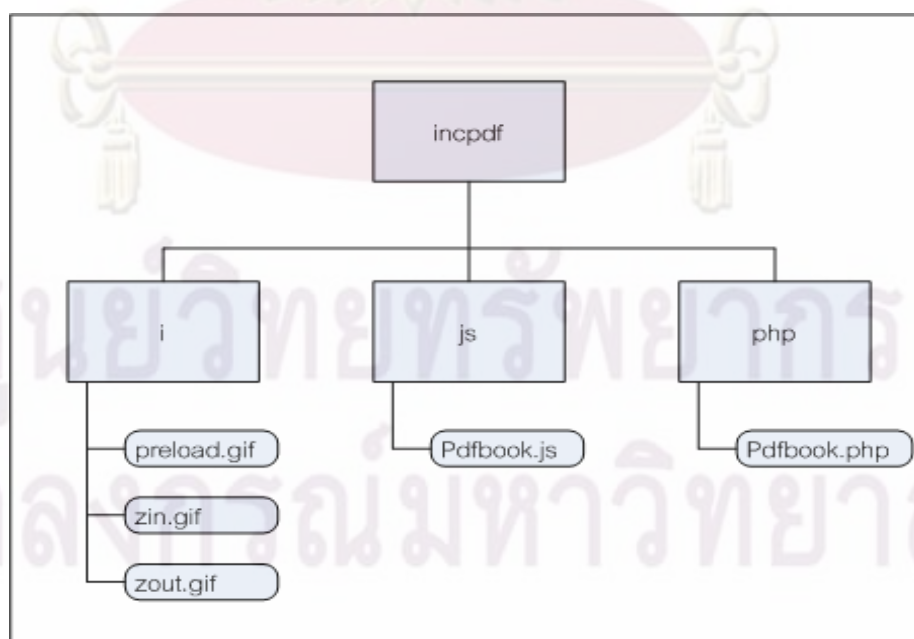
พัฒนาโปรแกรมบนเพิ่มที่มีชื่อว่า “pdfbook.js” เป็นเพิ่มจาวาสคริปต์ใช้สร้างวัตถุที่ใช้ทำงานเป็นระบบสนับสนุนบนเครื่องผู้ใช้ ลักษณะการเขียนโปรแกรมส่วนนี้เป็นการเขียนโปรแกรมแบบ Javascript Closure [10]

4.3.3 เพิ่มรูปภาพที่ใช้เป็นส่วนประกอบของเว็บ

- preload.gif เป็นเพิ่มภาพเคลื่อนไหวใช้สำหรับแสดงระหว่างการรอการค้นหาข้อมูล
- ⊕ zin.gif ใช้เป็นสัญลักษณ์ปุ่มในการขยายเอกสาร
- ⊖ zout.gif ใช้เป็นสัญลักษณ์ปุ่มในการย่อเอกสาร

4.3.4 โฟลเดอร์ที่ใช้ในการเก็บ

มีการจัดเก็บเพิ่มโปรแกรมให้อยู่รวมกันโดยมีการจัดเก็บดังรูปที่ 4.4

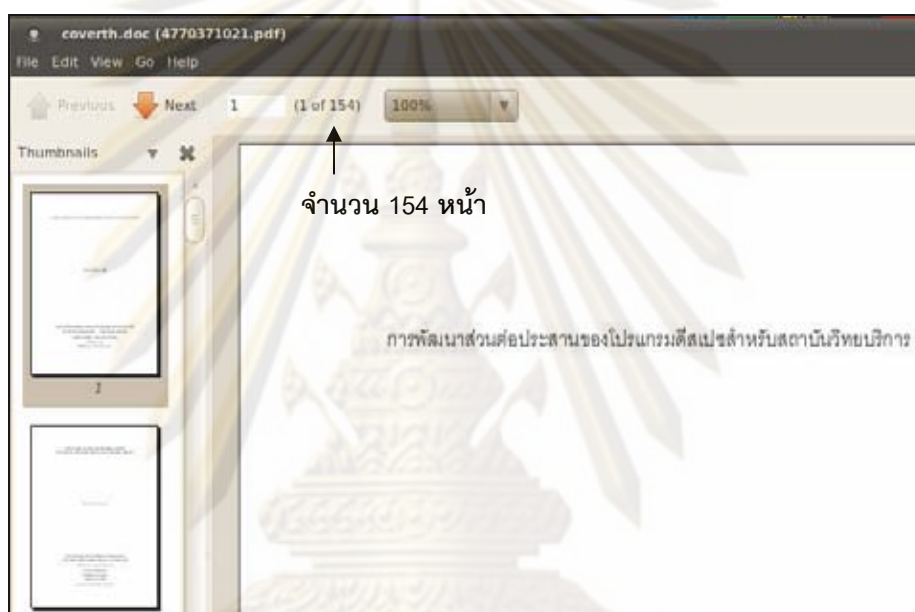


รูปที่ 4.4 โครงสร้างของโฟลเดอร์ในการจัดเก็บโปรแกรมของระบบ

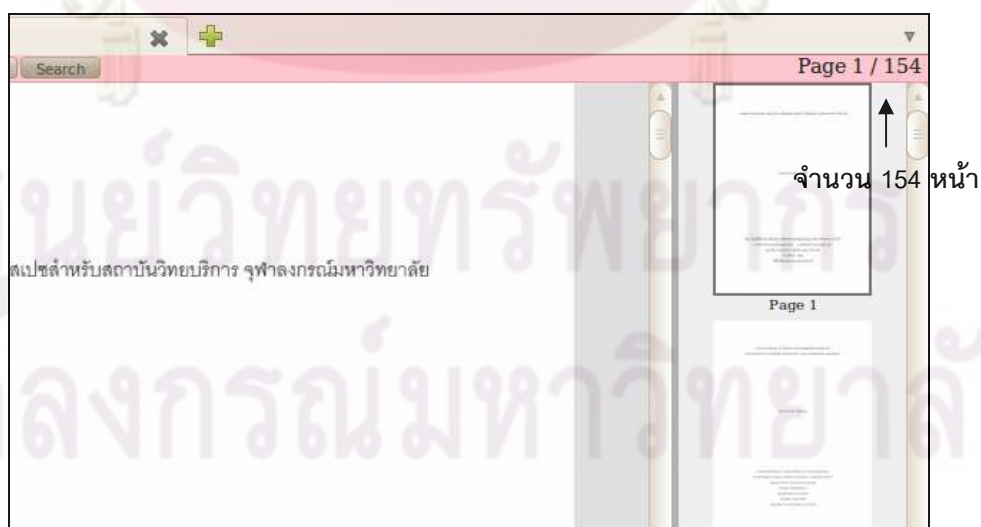
4.4 การทดสอบการทำงานของระบบ

4.4.1 การทดสอบการเปิดแฟ้มพีดีเอฟ

เนื่องจากในงานวิจัยนี้เป็นการพัฒนาระบบเพื่อทดลองใช้กับคลังปัญญาจุฬา ฯ เพื่อประเทศไทย ผู้วิจัยจึงได้ส่งมอบแฟ้มพีดีเอฟที่เก็บไว้ในคลังปัญญาจุฬา ฯ มาเพื่อใช้ในการทดสอบการใช้งานจำนวน 30 ฉบับ โดยมีการเปิดเทียบกันระหว่างระบบที่พัฒนาและโปรแกรมประยุกต์ที่ใช้สำหรับเปิดแฟ้มพีดีเอฟดังแสดงตัวอย่างในรูปที่ 4.5 และ 4.6



รูปที่ 4.5 ตัวอย่างการเปิดแฟ้มพีดีเอฟโดยใช้โปรแกรมประยุกต์



รูปที่ 4.6 ตัวอย่างการเปิดแฟ้มพีดีเอฟโดยใช้ระบบที่พัฒนาขึ้น

เห็นได้ว่าจำนวนหน้าที่ได้ตรงกัน โดยในการทดสอบใช้งานเปิดแฟ้มพีดีเอฟนั้น เมื่อระบบทำการนับจำนวนหน้าภายในแฟ้มพีดีเอฟได้จำนวนเท่าใด ก็จะสามารถเปิดหน้าเอกสารได้ทุกหน้าตามจำนวนที่นับได้ ซึ่งสรุปได้ว่าระบบมีความสามารถเปิดเอกสารแฟ้มพีดีเอฟได้ทุกฉบับ

4.4.2 การทดสอบการเปลี่ยนหน้าเอกสาร

สำหรับการเลื่อน Scroll Bar ในแนวขึ้นลงเพื่อต้องการแสดงเอกสารในหน้าอื่น ๆ จากการออกแบบระบบที่ได้กล่าวมาในบทที่ 3 นั้นมีการตรวจสอบการเลื่อนเปลี่ยนหน้าเอกสารเพื่อรับรองแฟ้มเจเพ็ทหน้าเอกสารนั้น โดยระบบจะมีการทำงานทุก ๆ 1 วินาทีเพื่อให้ได้เอกสารออกมาครบทุกหน้า จากการทดสอบระบบสามารถแสดงเอกสารได้ทุกหน้าเมื่อมีการเลื่อน แต่ในกรณีที่มีการเลื่อน Scroll Bar ขึ้นลงติดกันหลายครั้งโดยในแต่ละครั้งไปยังหน้าเอกสารที่ต่างกัน จะทำให้เกิดปรากฏการคอขวดขึ้น เนื่องจากได้มีการรับรองแฟ้มเจเพ็ทของเอกสารแต่ละหน้าอย่างต่อเนื่อง นั้นเท่ากับว่ามีการดาวน์โหลดแฟ้มเจเพ็ทหลายแฟ้มในเวลาเดียวกัน ซึ่งจะทำให้เกิดการรอการโหลดข้อมูลแต่ละแฟ้ม

4.4.3 การทดสอบการย่อขยายเอกสาร

จากขอบเขตข้อกำหนดในงานวิจัยนี้ให้สามารถย่อขยายเอกสารได้ในช่วง 10% - 800% ซึ่งได้มีการแบ่งเป็น 10 ช่วงคือ 10%, 25%, 50%, 75%, 80%, 100%, 150%, 200%, 400% และ 800% และได้ทำการทดสอบเปลี่ยนการย่อขยายในระดับต่าง ๆ พบว่าเมื่อมีการขยายเอกสารในระดับ 400% ขึ้นไปการดาวน์โหลดแฟ้มเจเพ็ทค่อนข้างช้าเนื่องจากมีขนาดของแฟ้มใหญ่ขึ้น แต่ยังคงอยู่ในระดับการใช้งานได้ปกติ และเมื่อปรับระดับขึ้นเป็น 800% ระบบจะไม่สามารถทำงานได้เลย เพราะทางฝั่งเครื่องแม่ข่ายจะต้องสร้างแฟ้มเจเพ็ทที่มีขนาดความกว้างเท่ากับ 5720 Pixel (715 Pixel x 8 เท่า) ซึ่งถือว่าเป็นขนาดที่ใหม่มาก ดังนั้นจึงได้กำหนดไว้ในส่วนของระบบการแปลงข้อมูลโดยมีเงื่อนไขว่า เมื่อต้องการเอกสารที่มีระดับการขยาย 800% ให้สร้างเอกสารที่มีความกว้างเท่ากับเอกสารขยายระดับ 400% และในขณะที่ระบบนำมาวางลงในช่องหน้าเอกสาร จะทำการดึงภาพเอกสารให้มีขนาดความกว้างเท่ากับ 5720 Pixel

4.4.4 การทดสอบการค้นหาข้อมูลภายในแฟ้มพีดีเอฟ

การค้นหาข้อมูลภายในแฟ้มพีดีเอฟตามที่ได้ออกแบบได้มีการแยกออกเป็น 2 ประเภทได้แก่ การค้นหาคำ และการค้นหาหัวข้อหลัก โดยจะสรุปผลการทดสอบดังนี้

4.4.4.1 การทดสอบการค้นหาคำภายในแฟ้มพีดีเอฟ

ผู้วิจัยได้ทำการทดสอบโดยใช้แฟ้มเอกสาร Microsoft Word (.doc) ซึ่งเป็นแฟ้มต้นฉบับของแฟ้มพีดีเอฟอีกแฟ้มหนึ่งที่จะนำมาทดสอบคือ filetest.doc และ filetest.pdf โดยที่นำแฟ้ม filetest.doc เปิดด้วยโปรแกรม Microsoft Word และแฟ้ม filetest.pdf เปิดด้วยระบบที่พัฒนาขึ้น จากนั้นทำการคำสั่งค้นหาคำจากโปรแกรมของทั้งสองแฟ้ม โดยใช้คำทดสอบเป็นจำนวน 10 คำ ได้แก่ การ, ชื่อ, คณะ, ทดสอบ, หน้า, รูปที่, ตาราง, สมการ, สรุป และค้นหา เมื่อทดสอบแล้วจึงได้ผลดังนี้

ตารางที่ 4.1 ผลการทดสอบการค้นหาคำ

คำที่ใช้ค้นหา	จำนวนคำที่ค้นใน filetest.doc	จำนวนคำที่ค้นใน filetest.pdf
การ	704	775
ชื่อ	308	306
คณะ	6	6
ทดสอบ	12	12
หน้า	31	0
รูปที่	67	67
ตาราง	47	47
สมการ	4	4
สรุป	6	6
ค้นหา	22	0
รวมจำนวนที่ค้น	1207	1223

จากตารางสรุปผลการค้นหาคำได้ว่าการมีความใกล้เคียงกันประมาณ 98.69% แต่จากข้อมูลที่ได้มีบางคำที่ไม่สามารถค้นหาได้เลยซึ่งถือเป็นส่วนผิดพลาดในการค้นหา

4.4.4.2 การทดสอบการค้นหาหัวข้อหลักภายในแฟ้มพีดีเอฟ

จากการทดสอบใช้การค้นหาหัวข้อหลักจากการแฟ้มพีดีเอฟที่นำมาทดสอบพบว่าข้อมูลผลลัพธ์ที่ได้เป็นข้อมูลที่มาจกหลาย ๆ ส่วนในเอกสารรวมกันอยู่โดยไม่ได้มีการจัดเรียง เนื่องจากเป็นกลุ่มข้อมูลที่ได้จากการอ่านรูปแบบของตัวอักษรที่มีตัวเลขระบุอยู่ จากการตรวจสอบข้อมูลที่ได้จากการค้นหานั้นยังมีข้อมูลบางส่วนที่ขาดหายไป เช่น หัวข้อหลักภายใน

เอกสารหายไปบางหัวข้อซึ่งถือได้ว่าในการค้นหาหัวข้อหลักนั้นยังไม่สามารถใช้งานได้ดี ทั้งนี้ในงานวิจัยนี้ต้องการที่จะสามารถเรียกดูข้อมูลเฉพาะที่เป็นรูปภาพ ซึ่งจากการค้นหาข้อมูลในส่วนนี้ได้ทำการค้นรายชื่อของกลุ่มรูปภาพที่มีอยู่ภายในแฟ้มพีดีเอฟ ทำให้สามารถที่จะหาตำแหน่งของข้อมูลที่เป็นรูปภาพได้

จากการทดสอบการค้นหาข้อมูลทั้ง 2 ประเภทนั้นจะสามารถทำการค้นหาข้อมูลจริงเฉพาะแฟ้มพีดีเอฟที่ถูกสร้างขึ้นจากการนำแฟ้มเอกสารที่มีตัวอักษรโดยเฉพาะการโปรแกรม Microsoft Word แต่หากมีการสร้างแฟ้มพีดีเอฟที่ได้จากการสแกนรูปภาพ รูปสร้างจากรูปภาพทั้งหมดระบบการค้นหาในส่วนนี้จะไม่สามารถค้นหาข้อมูลได้เลย

4.4.5 การทดสอบใช้งานกับเว็บเบราว์เซอร์

หลังจากได้ทดสอบใช้งานระบบต่าง ๆ แล้วจำเป็นที่จะต้องทดสอบการใช้งานของระบบที่พัฒนากับเว็บเบราว์เซอร์หลาย ๆ ยี่ห้อ ตามที่ได้กำหนดให้ขอบเขตงานวิจัย สามารถดูผลการทดลองใช้งานกับเว็บเบราว์เซอร์ต่าง ๆ ได้ในภาคผนวก ข ซึ่งสามารถใช้ได้กับทุกเว็บเบราว์เซอร์

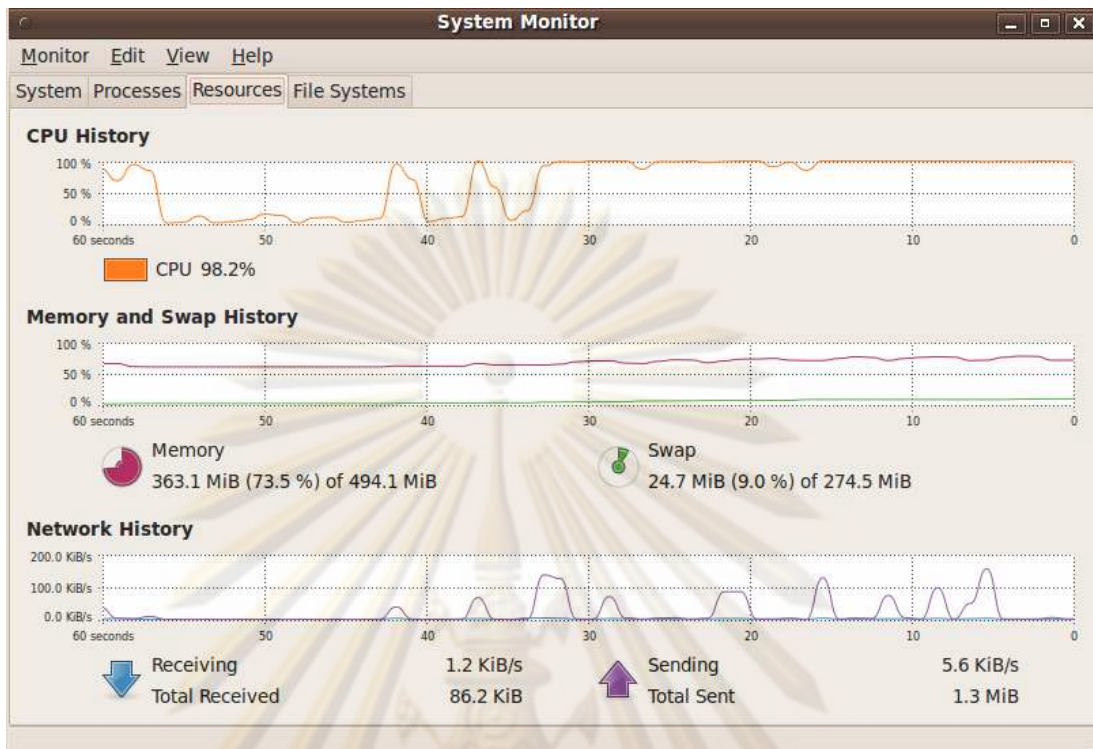
4.5 การวัดประสิทธิภาพของระบบ

ผู้วิจัยได้วัดประสิทธิภาพของระบบโดยใช้โปรแกรม System Monitor Version 2.28.0 ที่ติดตั้งมาพร้อมกับระบบปฏิบัติการ Ubuntu รุ่น 9.04 ที่ใช้บนเครื่องแม่ข่าย ซึ่งได้ทดสอบใช้งานระบบโดยมีผู้ใช้งาน 10 เครื่องโดยแบ่งวิธีการทดสอบเป็น 2 กรณีคือ

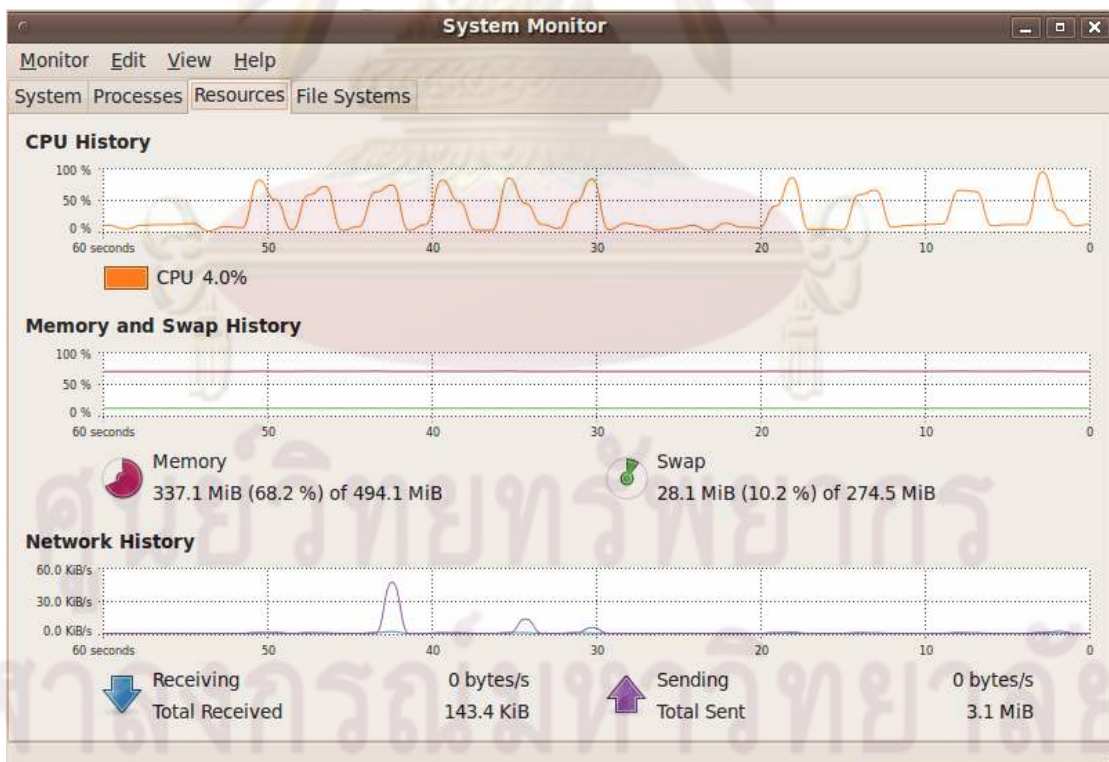
1. การทดสอบวัดประสิทธิภาพกรณีการเปิดใช้งานดูเอกสารตามปกติ
2. การทดสอบวัดประสิทธิภาพกรณีค้นหาข้อมูลภายในแฟ้มพีดีเอฟ

จากการวัดประสิทธิภาพการเปิดใช้งานดูเอกสารจะเห็นได้ว่าเมื่อมีการเริ่มใช้งานพร้อมกันหลายผู้ใช้เครื่องแม่ข่ายจะมีการทำงานในส่วนของหน่วยประมวลผล (CPU) 100% นั้นหมายถึงมีการใช้งานในการประมวลผลมาก แต่สำหรับด้านการใช้งานบนเครือข่ายจะมีการไหลข้อมูลเป็นช่วง ๆ ไม่เกิน 2 – 3 วินาที นั้นหมายถึงเป็นช่วงระยะเวลาที่หน่วยประมวลผลได้ทำการแปลงแฟ้มพีดีเอฟเป็นแฟ้มเจพีคเสิร์ฟต่อ 1 หน้า ทั้งนี้ขึ้นอยู่กับการใช้งานของผู้ใช้แต่ละคนว่าจะมีการใช้งานเปลี่ยนหน้าเอกสารอย่างต่อเนื่องหรือไม่ ก็จะมีผลในการทำงานของระบบที่สูงเช่นเดียวกันดังรูปที่

4.7



รูปที่ 4.7 ผลการวัดประสิทธิภาพของระบบ



รูปที่ 4.8 ผลการวัดประสิทธิภาพของระบบในขณะที่ค้นหาข้อมูล

จากรูปที่ 4.8 เป็นการวัดประสิทธิภาพการค้นหาข้อมูล จะเห็นได้ว่าจะมีส่วนการใช้งานมากเฉพาะหน่วยประมวลผลที่มีการใช้งานเป็นช่วง ๆ ซึ่งในขณะเดียวกันการใช้งานด้านเครือข่ายจะน้อยมาก เนื่องจากการส่งข้อมูลผลการค้นหาเป็นข้อมูลชนิดตัวอักษรซึ่งมีขนาดของข้อมูลน้อยมาก จึงทำให้มีการใช้ทรัพยากรส่วนนี้ไม่มาก

จากการวัดประสิทธิภาพทั้ง 2 กรณีนั้นจะเห็นได้ว่ามีการใช้ทรัพยากรของหน่วยประมวลผลบนเครื่องแม่ข่ายมาก ผู้วิจัยจึงได้ทำการทดสอบเขียนโปรแกรมเพื่อวัดความเร็วของการทำงานหน่วยประมวลผลในการแปลงแฟ้มพีดีเอฟเป็นแฟ้มเจพีคใน 1 หน้าเอกสารซึ่งได้ผลดังรูปที่ 4.9 ซึ่งเฉลี่ยเวลาได้ประมาณ 1 วินาที ดังตารางที่ 4.2

ตารางที่ 4.2 ผลการวัดความเร็วในการแปลงเอกสาร

เอกสารหน้าที่	เวลาที่ใช้ (วินาที)	เอกสารหน้าที่	เวลาที่ใช้ (วินาที)
1	1.085462808609	16	1.0665979385376
2	1.0910241603851	17	1.0217339992523
3	1.0662319660187	18	1.1050620079041
4	1.0494849681854	19	1.097244977951
5	0.9870572090148	20	1.0815379619598
6	1.1284379959106	21	1.0774099826813
7	1.0830891132355	22	1.0787091255188
8	1.06924700737	23	1.0987348556519
9	1.0684750080109	24	1.0717549324036
10	1.0660519599915	25	1.0769288539886
11	1.0655119419098	26	1.0685038566589
12	1.070631980896	27	1.0612409114838
13	1.0754759311676	28	1.0633821487427
14	1.0823459625244	29	1.064342975616
15	1.0314779281616	30	1.0496799945831

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากการพัฒนาระบบสนับสนุนสำหรับโปรแกรมแสดงผลแฟ้มพีดีเอฟแบบออนไลน์ ในรูปแบบแฟ้มเจพีก ซึ่งได้มีการนำมาทดสอบใช้งานร่วมกับคลังปัญญาจุฬาฯ เพื่อประเทศไทย ซึ่งนำมาใช้งานแทนโปรแกรมประยุกต์สำหรับเปิดเอกสารวิทยานิพนธ์ในรูปแบบแฟ้มพีดีเอฟ โดยมีโปรแกรมที่สามารถแปลงแฟ้มพีดีเอฟให้อยู่ในรูปแบบแฟ้มเจพีกได้โดยผ่านระบบที่ได้พัฒนาขึ้นในการใช้งานแต่ละครั้งสามารถเปิดอ่านข้อมูลภายในแฟ้มพีดีเอฟได้ทันที และยังได้ทำระบบในการลงทะเบียนให้ผู้ใช้สมัครเป็นสมาชิกเพื่อใช้ในการยืนยันการมีตัวตนของผู้ใช้เพื่อจำกัดจำนวนหน้าในการอ่าน โดยสมาชิกสามารถอ่านแฟ้มพีดีเอฟได้ทั้งหมด ในขณะที่ผู้ใช้ทั่วไปสามารถเปิดเอกสารอ่านได้เพียง 5 หน้าแรกเท่านั้น สำหรับการใช้งานทั่วไปผู้ใช้สามารถทำการเลือกดูหน้าอื่น ๆ ของเอกสารได้โดยการเลื่อน Scroll Bar ของหน้าเอกสารไปยังตำแหน่งหน้าที่ต้องการ อีกทั้งยังสามารถปรับขนาดของเอกสารโดยการเลือกปุ่มย่อหรือขยายเอกสารได้ซึ่งมีค่าอยู่ระหว่าง 10% - 800%

สำหรับการค้นหาค่าภายในแฟ้มพีดีเอฟสามารถทำได้โดยการระบุค่าที่ต้องการค้นหาลงให้ช่องค้นหาที่อยู่บนส่วนต่อประสานผู้ใช้และเริ่มทำการค้น ผลระบบก็จะแสดงผลที่ค้นหาได้ในรูปแบบ Drop-down List เพื่อสามารถให้ผู้ใช้เลือกรายการของค่าที่ค้นหาเพื่อเปลี่ยนหน้าเอกสาร ปัจจุบันไปยังตำแหน่งของหน้าที่ทำการเลือกได้ทันที และการค้นหาหัวข้อหลักของเอกสารนั้นผลที่ได้ยังไม่สามารถแสดงข้อมูลออกมาได้ครบทั้งหมด ซึ่งการค้นหาข้อมูลนั้นได้ใช้วิธีการแปลงแฟ้มพีดีเอฟให้อยู่ในรูปแบบข้อมูลตัวอักษรซึ่งเป็นข้อมูลที่เป็นเนื้อหาภายในแฟ้มพีดีเอฟ

สามารถนำระบบที่พัฒนาไปติดตั้งใช้งานร่วมกับระบบอื่นที่ต้องการนำเสนอเอกสารชนิดแฟ้มพีดีเอฟได้โดยที่ระบบนั้นจะต้องมีคุณสมบัติของเครื่องแม่ข่ายดังที่กล่าวไว้ในบทที่ 4 เรื่องของซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ ซึ่งในการใช้งานสามารถนำไปรวมกับหน้าเว็บไซต์ที่มีอยู่โดยแทรกส่วนที่เป็นระบบแสดงผลเข้าไปทำงานร่วมกัน

ในการทดสอบการใช้งานและวัดประสิทธิภาพโดยดูจากการทำงานในส่วนเครื่องแม่ข่ายพบว่ามีการใช้งานที่สูงในช่วงระยะเวลาที่มีการแปลงข้อมูล ซึ่งการจัดหาเครื่องคอมพิวเตอร์ที่จะนำมาเป็นเครื่องแม่ข่ายควรเป็นเครื่องแม่ข่ายที่มีความเร็วของหน่วยประมวลผลที่สูง

5.2 ข้อจำกัด

งานวิจัยนี้ได้ถูกออกแบบมาให้สำหรับอ่านแฟ้มพีดีเอฟที่มีอยู่ภายในระบบเดียวกัน และสามารถเรียกถึงแฟ้มได้โดยตรง ไม่เหมาะสมกับการใช้งานร่วมกับเครื่องแม่ข่ายที่ใช้เก็บแฟ้มพีดีเอฟที่มีเครือข่ายห่างไกลกัน โดยเฉพาะเมื่อใช้อ่านแฟ้มพีดีเอฟที่มีขนาดใหญ่มากจะทำให้การทำงานล่าช้าหรือเกิดข้อผิดพลาดในการแสดงผลได้

อีกทั้งเรื่องของการค้นหาคำที่มีการแสดงผลที่ไม่สะดวก และในเรื่องของการค้นหาข้อมูลยังไม่สามารถค้นหาตำแหน่งของคำได้บนหน้าเอกสารนั้น แต่จะบอกได้เฉพาะว่าเป็นหน้าใดเท่านั้น ซึ่งในการอ่านแฟ้มพีดีเอฟและต้องการค้นหาข้อมูลนั้นจะต้องใช้แฟ้มพีดีเอฟที่สร้างจากแฟ้มเอกสารที่ประกอบด้วยตัวอักษรเท่านั้นจึงจะสามารถค้นหาข้อมูลได้

5.3 ปัญหาในการใช้งาน

ในกรณีใช้เบราว์เซอร์ Firefox ที่เปิดการใช้งานของระบบที่มีเครือข่ายความเร็วต่ำ ซึ่งในช่วงแรกของการเปิดเอกสารการจัดเรียงหน้าเอกสารอาจเกิดการทับซ้อนกับระหว่างเอกสารได้ ซึ่งถ้าเกิดกรณีนี้ให้ผู้ใช้ทำการ Refresh ระบบใหม่อีกครั้ง

5.4 แนวทางการวิจัยต่อ

จากข้อจำกัดในเรื่องของการค้นหาและแสดงผลได้ไม่ดีพอ จึงเป็นแนวทางหนึ่งสำหรับการพัฒนาให้ระบบความแสดงผลการค้นหาให้มีความสะดวกมากขึ้น ซึ่งจำเป็นที่จะต้องศึกษาเรื่องการใช้การรู้จำอักขระ OCR (Optical Character Recognition) เพื่อค้นหาและแสดงตำแหน่งของคำที่ค้นหาเพื่อทำการแสดงตำแหน่งคำให้ชัดเจน

การพัฒนาระบบสนับสนุนให้สามารถแสดงได้บนเบราว์เซอร์ที่มีอยู่บนโทรศัพท์เคลื่อนที่ (Mobile Phone) ซึ่งจำเป็นที่จะต้องมีการศึกษาการทำงานบนเบราว์เซอร์เหล่านั้นด้วย

รายการอ้างอิง

- [1] สุรพงษ์ เชาว์เชียวชาญ. การออกแบบและพัฒนาส่วนจำเพาะการค้นข้อความไทยในเอกสารพีดีเอฟ, วิทยานิพนธ์ปริญญาามหาบัณฑิต, สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2543.
- [2] Adobe Systems Incorporated. Adobe PostScript 3 [Online]. 2009. Available from: <http://www.adobe.com/products/postscript/> [2010, March 1]
- [3] ลัญฉกร วุฒิสีทธิกุลกิจ และคณะ. มาตรฐานไฟล์ภาพ JPEG. เทคโนโลยีการบีบอัดข้อมูลเบื้องต้น. หน้า 147 - 169. กรุงเทพฯ : สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย, 2549.
- [4] ImageMagick Studio LLC. ImageMagick [Online]. Available from: <http://www.imagemagick.org/script/index.php> [2008, March 19]
- [5] W3Schools. JavaScript [Online]. Available from: <http://www.w3schools.com/js/default.asp> [2008, March 19]
- [6] Wikipedia. AJAX [Online]. Available from: <http://th.wikipedia.org/wiki/เอแจ็กซ์> [2010, March 1]
- [7] The World Wide Web Consortium (W3C). Cascading Style Sheets [Online]. Available from: <http://www.w3.org/Style/CSS/> [2010, March 1]
- [8] The World Wide Web Consortium (W3C). Document Object Model (DOM) [Online]. Available from: <http://www.w3.org/Style/CSS/> [2010, March 1]
- [9] Wikipedia. XMLHttpRequest [Online]. Available from: <http://th.wikipedia.org/wiki/XMLHttpRequest> [2010, March 1]
- [10] Richard Cornford. Javascript Closures [Online]. 2004. Available from: http://www.jibbering.com/faq/faq_notes/closures.html [2010, March 1]
- [11] Wikipedia. ภาษา PHP [Online]. Available from: <http://th.wikipedia.org/wiki/PHP> [2010, March 1]
- [12] the GNU General Public License (GPL). XPDF [Online]. Available from: <http://www.foolabs.com/xpdf/>. [2008, March 19]
- [13] Wikipedia. Regular Expression [Online]. Available from: http://en.wikipedia.org/wiki/Regular_expression. [2010, March 1]



ภาคผนวก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก.

คำอธิบายยูสเคส

จากรูปที่ 3.1 แผนภาพยูสเคสการวิเคราะห์ระบบ สามารถแสดงรายละเอียดด้วยคำอธิบายยูสเคสดังตารางที่ ก.1 – ก.10

ตารางที่ ก.1 คำอธิบายยูสเคส Insert Form

ชื่อยูสเคส : Insert Form
ผู้กระทำหลัก : ผู้ใช้
รายละเอียด : เพื่อใช้รับข้อมูลจากผู้ใช้สำหรับลงทะเบียน หรือเข้าสู่ระบบ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. รับข้อมูลจากผู้ใช้ 2. ส่งข้อมูลยังเครื่องแม่ข่าย

ตารางที่ ก.2 คำอธิบายยูสเคส Register System

ชื่อยูสเคส : Register System
ผู้กระทำหลัก : ระบบ
รายละเอียด : บันทึกข้อมูลการลงทะเบียนจากผู้ใช้เข้าสู่ระบบ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. บันทึกข้อมูลลงทะเบียนเข้าสู่ระบบ 2. แจ้งผลการลงทะเบียนของผู้ใช้

ตารางที่ ก.3 คำอธิบายยูสเคส Login System

ชื่อยูสเคส : Login System
ผู้กระทำหลัก : ระบบ
รายละเอียด : ตรวจสอบการเป็นสมาชิกของผู้ใช้ และยืนยันการเข้าสู่ระบบ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. ตรวจสอบข้อมูลการเป็นสมาชิก 2. แจ้งผลการเข้าสู่ระบบ
เงื่อนไข : <ol style="list-style-type: none"> 2a. ข้อมูลสมาชิกถูกต้องยืนยันการเข้าสู่ระบบ 2b. ข้อมูลไม่ถูกต้องให้กลับไปเข้าสู่ระบบใหม่อีกครั้ง

ตารางที่ ก.4 คำอธิบายยูสเคส View PDF

ชื่อยูสเคส : View PDF
ผู้กระทำหลัก : ผู้ใช้
รายละเอียด : เป็นส่วนต่อประสานผู้ใช้หลักของระบบการแสดงผลแฟ้มพีดีเอฟ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. ผู้ใช้เลือกอ่านแฟ้มพีดีเอฟ 2. สร้างระบบพื้นฐานสำหรับรองรับการแสดงผลแฟ้มพีดีเอฟ 3. เริ่มทำการเรียกแฟ้มเจพีกจากเครื่องแม่ข่าย 4. นำแฟ้มเจพีกที่ได้วางลงตำแหน่งของหน้าเอกสาร

ตารางที่ ก.5 คำอธิบายยูสเคส Count Page

ชื่อยูสเคส : Count Page
ผู้กระทำหลัก : ระบบ
รายละเอียด : ทำหน้าที่ในการนับจำนวนหน้าของแฟ้มพีดีเอฟ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. อ่านแฟ้มพีดีเอฟที่ผู้ใช้ต้องการอ่านเพื่อับจำนวนหน้า 2. ตรวจสอบ SESSION ในระบบว่ามีการเข้าสู่ระบบของผู้ใช้หรือไม่ 3. ส่งค่าตัวเลขจำนวนหน้าของแฟ้มพีดีเอฟ
เงื่อนไข : <ol style="list-style-type: none"> 2a. พบว่าผู้ใช้ได้เข้าสู่ระบบแล้วให้ผลการนับจำนวนหน้าตามที่ได้ 2b. พบว่าผู้ใช้ไม่ได้ทำการเข้าสู่ระบบให้ผลการนับเท่ากับ 5

ตารางที่ ก.6 คำอธิบายยูสเคส Render Page

ชื่อยูสเคส : Render Page
ผู้กระทำหลัก : ระบบ
รายละเอียด : ทำหน้าที่แปลงแฟ้มพีดีเอฟเป็นแฟ้มเจเพ็ท
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. รับคำสั่งการร้องขอข้อมูลเอกสารหน้าที่ n 2. ทำการแปลงแฟ้มพีดีเอฟเป็นแฟ้มเจเพ็ทหน้าที่ n 3. ส่งข้อมูลแฟ้มเจเพ็ทตามที่ร้องขอ

ตารางที่ ก.7 คำอธิบายยูสเคส Scroll Page

ชื่อยูสเคส : Scroll Page
ผู้กระทำหลัก : ผู้ใช้
รายละเอียด : เป็นการเคลื่อนหน้าเอกสารในแนวขึ้นลง เพื่อเปลี่ยนไปดูเอกสารหน้าอื่น
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. ผู้ใช้ทำการลาก Scroll Bar ของหน้าเอกสารเพื่อเปลี่ยนไปดูหน้าอื่น 2. ระบบตรวจพบการเปลี่ยนหน้า จึงร้องขอข้อมูลเอกสารหน้าใหม่ 3. รับข้อมูลเอกสารหน้าใหม่ไปวางลงในช่องของหน้านั้น ๆ

ตารางที่ ก.8 คำอธิบายยูสเคส Zoom PDF Page

ชื่อยูสเคส : Zoom PDF Page
ผู้กระทำหลัก : ผู้ใช้
รายละเอียด : ระบบเปลี่ยนขนาดของเอกสารโดยกำหนดให้อยู่ในช่วง 10 – 800% มี 10 ระดับ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. ผู้ใช้คลิกเปลี่ยนขนาดเอกสารบนระบบแสดงผลแฟ้มพีดีเอฟ 2. ระบบทำการล้างหน้าเอกสารเดิมทั้งหมด 3. เริ่มการสร้างหน้าเอกสารชุดใหม่โดยมีขนาดตามที่ได้เลือกไว้

ตารางที่ ก.9 คำอธิบายยูสเคส Search Page

ชื่อยูสเคส : Search Page
ผู้กระทำหลัก : ผู้ใช้
รายละเอียด : ผู้ใช้สามารถทำการค้นหาข้อมูลที่อยู่ภายในแฟ้มพีดีเอฟได้
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. ผู้ใช้คลิกเลือกประเภทการค้นหาข้อมูล 2. ทำการลงข้อมูลค้นหาไปยังเครื่องแม่ข่าย 3. รับผลการค้นหาที่ได้แสดงใน Drop-Down List 4. ผู้ใช้คลิกดูข้อมูลที่ค้นหาจาก Drop-Down List 5. หน้าเอกสารทำการเคลื่อนไปยังหน้าเอกสารที่มีข้อมูลที่ค้นหา

ตารางที่ ก.10 คำอธิบายยูสเคส Search Data

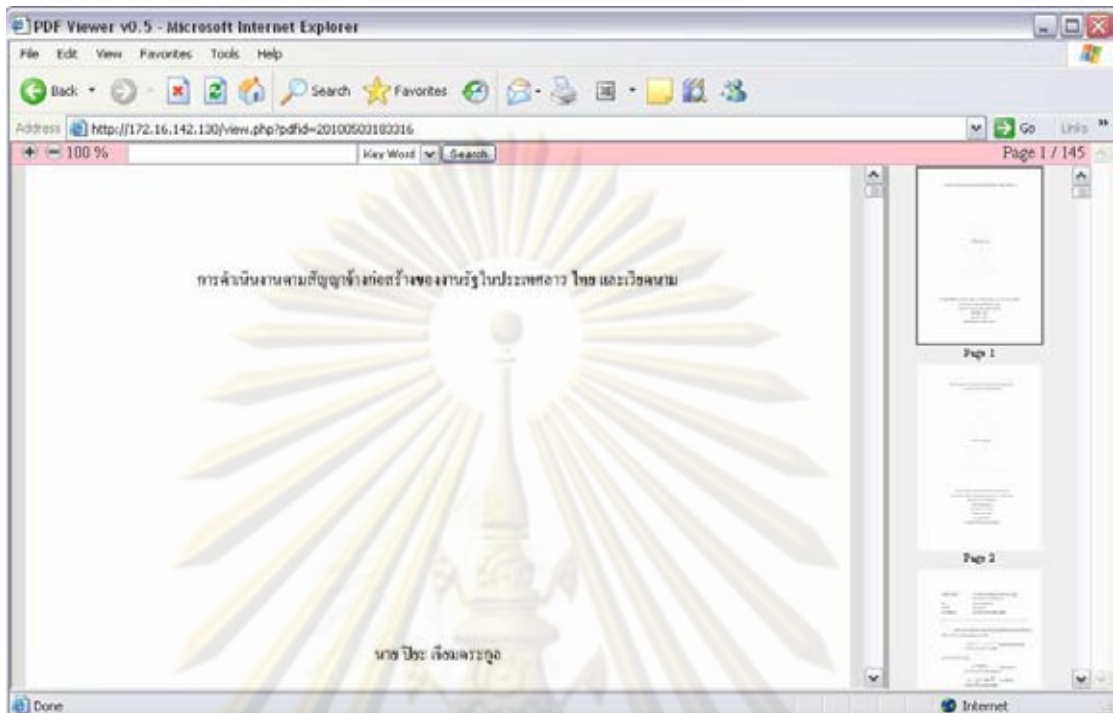
ชื่อยูสเคส : Search Data
ผู้กระทำหลัก : ระบบ
รายละเอียด : รับคำสั่งและทำการค้นหาข้อมูลที่อยู่ภายในแฟ้มพีดีเอฟ และส่งข้อมูลกลับ
ขั้นตอนการทำงานหลัก : <ol style="list-style-type: none"> 1. รับคำสั่งการค้นหาข้อมูล 2. ทำการค้นหาข้อมูลคำ หัวข้อหลัก ในแฟ้มพีดีเอฟ 3. จัดเรียงข้อมูลที่ได้จากการค้นเป็นรายการระบุถึงตำแหน่งของหน้าเอกสารนั้น 4. ส่งข้อมูลกลับเพื่อเพื่อผล

ภาคผนวก ข

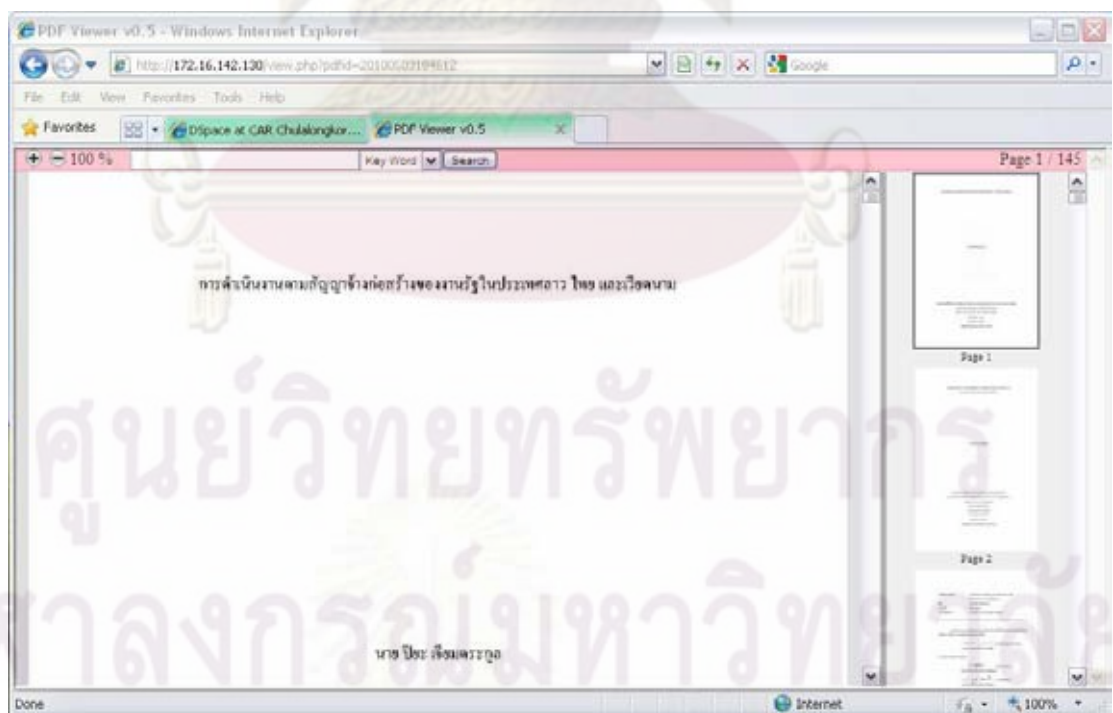
ตัวอย่างการใช้งานระบบบนเว็บไซต์ต่าง ๆ



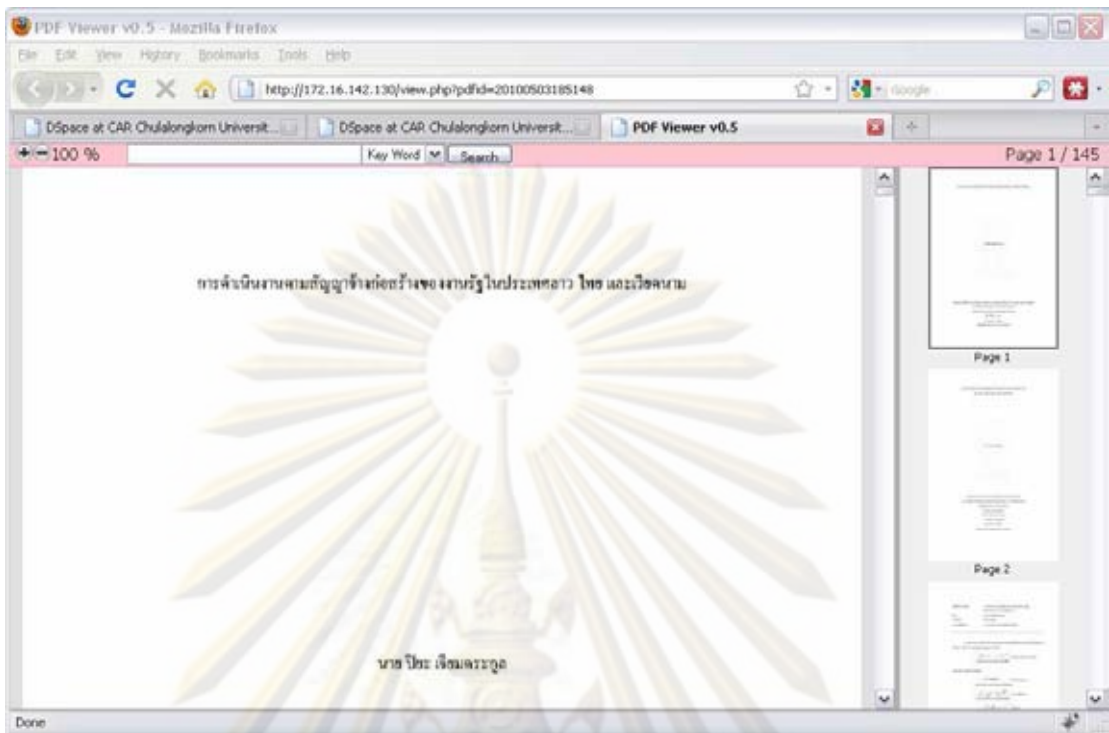
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



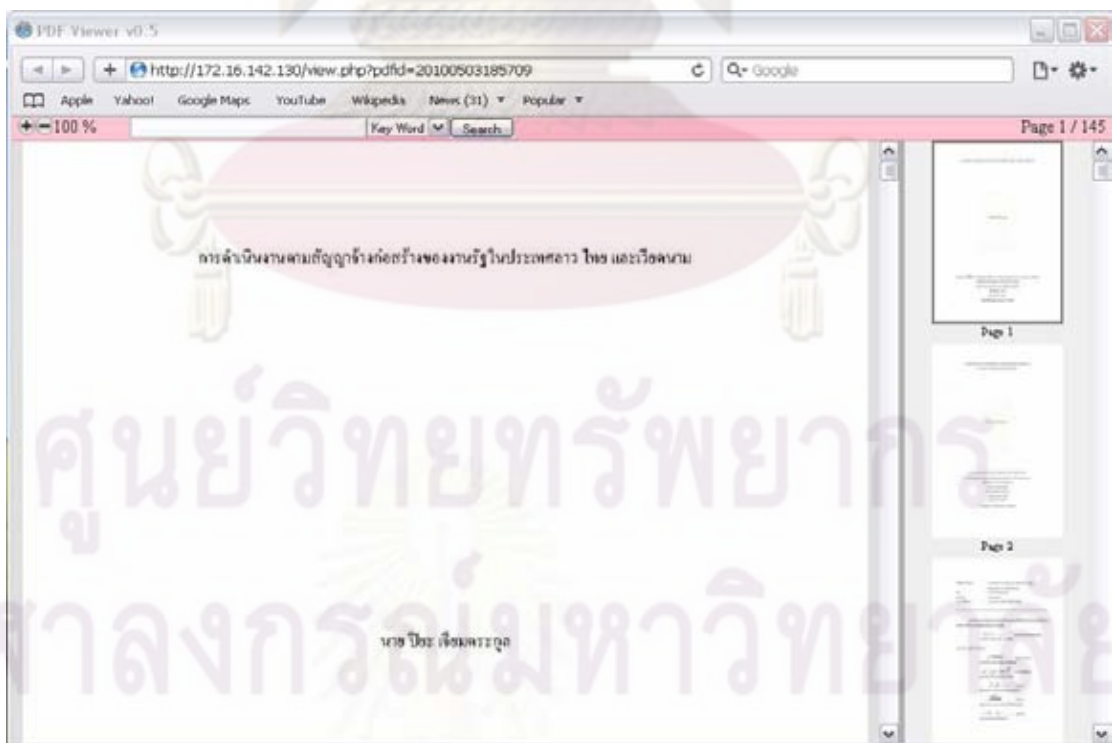
รูปที่ ข.1 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Internet Explorer 6



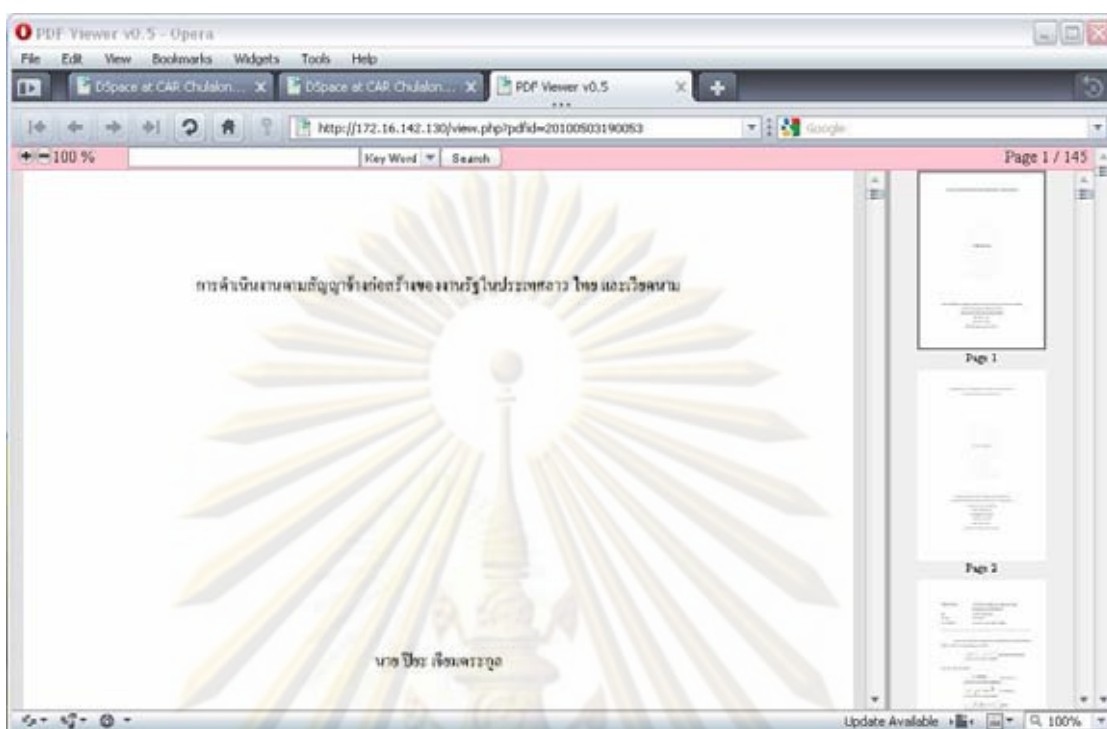
รูปที่ ข.2 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Internet Explorer 7



รูปที่ ข.3 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Mozilla Firefox



รูปที่ ข.4 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Apple Safari



รูปที่ ข.5 ตัวอย่างการใช้งานระบบแสดงผลแฟ้มพีดีเอฟบน Opera

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค

บทความวิชาการ



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ระบบการอ่านเอกสารอิเล็กทรอนิกส์หลากหลายหน้าแบบออนไลน์ Online Multi-Page Electronic Document Reader System

อัสวิน วงษ์แก่นคำ วิวัฒน์ วัฒนาวุฒิ

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปทุมวัน กทม. 10330

e-mail : Ussawin.V@student.chula.ac.th, Wiwat@chula.ac.th

บทคัดย่อ

การอ่านเอกสารอิเล็กทรอนิกส์ "แฟ้มพีดีเอฟ" ที่มีความต้องการจำกัดจำนวนการแสดงผลเนื้อหาหรือต้องการให้สามารถอ่านเนื้อหาได้เพียงบางส่วน และรวมไปถึงการแสดงผลข้อมูลภายในแฟ้มพีดีเอฟนั้นได้ทันที โดยแม้ว่าเอกสารจะมีขนาดใหญ่มากตาม งานวิจัยนี้จะศึกษาและพัฒนาวิธีการแสดงผลข้อมูลบนแฟ้มพีดีเอฟเป็นแบบเพิ่มรูปภาพ (Image File) ชนิดเจเพ็ก (JPEG) เพื่อนำมาแสดงให้กับผู้ใช้ผ่านระบบอินเทอร์เน็ต โดยแสดงผลบนเว็บเบราว์เซอร์ โดยไม่จำเป็นต้องใช้โปรแกรมประยุกต์สำหรับการแสดงผลข้อมูลภายในแฟ้มพีดีเอฟ

คำสำคัญ: โปรแกรมค้นดูแฟ้มพีดีเอฟแบบออนไลน์
ห้องสมุดเสมือน

Abstract

Displaying PDF file on the internet required completely content file sent to user that is not flexible method. Partial content displaying technic will help user to access to direct information immediately although the file size is very huge. In this paper, we propose development PDF displaying online system by image translation approach in order to user comfortable. No application need to display information except only general browser.

Keyword: Online PDF Browser, Digital Library

1. บทนำ

เอกสารอิเล็กทรอนิกส์แบบพีดีเอฟ (PDF - Portable Document Format) เริ่มมีความนิยมแพร่หลายมากขึ้นเนื่อง

ด้วยความสะดวกในหลายประการเช่น การจัดเก็บ เพราะเป็นเพียงแฟ้มข้อมูลไม่ต้องใช้เนื้อที่เหมือนหนังสือจริง ง่ายต่อการเสาะหาเพราะสามารถดาวน์โหลดออนไลน์ผ่านทางอินเทอร์เน็ตได้ การพกพาอุปกรณ์อิเล็กทรอนิกส์ปัจจุบันมีอุปกรณ์พกพาหลายประเภทสามารถแสดงผลเอกสารอิเล็กทรอนิกส์ได้อย่างเช่น โทรศัพท์มือถือ พีดีเอ (PDA) คอมพิวเตอร์พกพา (Laptop) ทำให้สามารถอ่านเอกสารได้ในทุก ๆ ที่ แต่การที่จะดาวน์โหลดเอกสารมานั้น อาจจะเป็นการลำบากถ้าหากเอกสารนั้นมีขนาดของข้อมูลที่ใหญ่มาก ทำให้ต้องเสียเวลาคดาวน์โหลด และเมื่อดาวน์โหลดมาแล้วเอกสารนั้นก็ยังมีเนื้อหาที่ไม่ตรงตามความต้องการของผู้ใช้ ทำให้เสียเวลาโดยใช่เหตุ

การอ่านเอกสารบางประเภทเช่นเอกสารงานวิจัย หรือวิทยานิพนธ์นั้นไม่ต้องการให้ผู้ใช้สามารถอ่านข้อมูลในเอกสารได้ทั้งหมด คือต้องการให้อ่านได้เพียงบางส่วนของเอกสารนั้น ๆ หรือต้องการให้อ่านทั้งหมดได้แต่ไม่ต้องการให้สามารถดาวน์โหลดเอกสารนั้นไปได้ แต่เนื่องจากคุณสมบัติของเอกสารอิเล็กทรอนิกส์ หรือในที่นี้จะเรียกว่า "แฟ้มพีดีเอฟ" มีลักษณะเป็นแฟ้มเอกสารที่รวบรวมข้อมูลของเอกสารทุกอย่างไว้ในแฟ้ม ๆ เดียว ซึ่งถ้าผู้ใช้ต้องการอ่านแฟ้มพีดีเอฟที่อยู่ในระบบอินเทอร์เน็ต จะต้องทำการดาวน์โหลดแฟ้มพีดีเอฟมาก่อนทั้งหมดแล้วจึงสามารถเปิดอ่านเอกสารได้ตามที่ได้กล่าวมาแล้ว และการอ่านนั้นจะอ่านข้อมูลในเอกสารได้ทั้งหมดทุกหน้า โดยที่ผู้ที่ให้บริการ หรือเจ้าของแฟ้มพีดีเอฟนั้นไม่สามารถจำกัดจำนวนหน้าในการอ่านได้เลย

การแสดงผลเนื้อหาของเอกสารสำหรับในงานวิจัยนี้จะกล่าวถึงแฟ้มเอกสารอิเล็กทรอนิกส์แบบ พีดีเอฟเพียงอย่าง

เดียว ซึ่งจากเหตุผลต่าง ๆ ที่ได้กล่าวมาแล้วนั้น จึงเป็นปัญหาสำคัญในการจำกัดเนื้อหาในการอ่าน และป้องกันการดาวน์โหลดเพิ่มพืดีโอฟ โดยในปัจจุบันถ้าต้องการให้ผู้ใช้อ่านเพิ่มพืดีโอฟ โดยจำกัดจำนวนเนื้อหาตามที่ต้องการจะต้องทำการสำเนาเพิ่มขึ้นมาใหม่แล้วตัดข้อมูลไปเพิ่มให้เหลือเฉพาะส่วนเนื้อหาที่ต้องการให้อ่านได้เท่านั้น ซึ่งวิธีนี้จะเสียเวลาและเสียเนื้อที่ในระบบโดยใช่เหตุและยังทำให้การจัดการเพิ่มเป็นเรื่องยุ่งยากอีก เอกสารเรื่องเดียวกันเดิมเพิ่มมากกว่า 1 เพิ่ม

งานวิจัยนี้จะศึกษาและพัฒนาวิธีการสนับสนุนการแสดงผลข้อมูลภายในเพิ่มพืดีโอฟที่อยู่บนเครื่องแม่ข่าย ด้วยวิธีการทำเป็นเพิ่มรูปภาพ (Image File) ชนิดเจพีก (JPEG) เพื่อนำมาแสดงให้กับผู้ใช้ผ่านระบบอินเทอร์เน็ต โดยแสดงบนเว็บเบราว์เซอร์

2. วรรณกรรมที่เกี่ยวข้อง

2.1 โปสต์สคริปต์ (PostScript) [2]

“ภาษา PostScript” ซึ่งจัดเป็นภาษาประเภท Page Description Language (PDL) คือใช้บรรยายสิ่งที่จะให้พิมพ์ลงบนแต่ละหน้ากระดาษ (Page) โปสต์สคริปต์ได้มีการแบ่งออกเป็น 3 ช่วงคือ PostScript level 1, PostScript level 2, PostScript 3

1) 1984 : PostScript level 1

ในปี 1984 PostScript ได้เปิดตัวออกมาเป็นครั้งแรก โดยใช้ชื่อว่า “PostScript” ส่วนคำว่า “Level 1” นั้น เพิ่มขึ้นมาภายหลัง เพื่อให้เห็นความแตกต่างจากรุ่นใหม่ ที่ออกมาทีหลัง ซึ่งเรียกว่าเป็น “Level 2” โปสต์สคริปต์ถือเป็นภาษาที่มีความสามารถมาก นับตั้งแต่ช่วงเวลาที่เริ่มต้นที่ โปสต์สคริปต์ยังต้องการระบบที่มีความสามารถมาก ในการรองรับการทำงาน ซึ่งในความเป็นจริงแล้ว ช่วงระยะปีแรก ๆ เครื่องพิมพ์ที่ใช้โปสต์สคริปต์มีความสามารถมากจนแม้แต่ Macintosh เอง ยังต้องนำ โปสต์สคริปต์เข้ามาใช้กับงานด้านการเตรียมพิมพ์ หรือเรียกพิมพ์ด้วยคอมพิวเตอร์ (Electronic prepress)

ซึ่งข้อได้เปรียบของโปสต์สคริปต์ที่ระบบอื่น ๆ ไม่มีได้แก่ โปสต์สคริปต์ไม่ขึ้นอยู่กับอุปกรณ์ตัวใดตัวหนึ่ง โดยเฉพาะ (Device independent) นั้นหมายความว่าลักษณะการพิมพ์ต่าง ๆ ที่กำหนดไว้ในภาษาโปสต์สคริปต์จะให้ผลลัพธ์ที่เหมือนกันบนอุปกรณ์ที่ใช้โปสต์สคริปต์ตัวใดก็ได้

ไม่ว่าจะเป็นเครื่องพิมพ์เลเซอร์ ที่มีความละเอียดเป็น 300 dpi (จุดต่อนิ้ว หรือ dot per inch) หรือ 600 dpi หรือเครื่องเรียงพิมพ์ระดับมืออาชีพ ที่มีความละเอียดเป็น 2400 dpi หรือสูงกว่านั้น ผู้ผลิตบางรายสามารถซื้อลิขสิทธิ์ (License) ตัวแปลภาษาหรืออินเทอร์พรีเตอร์ (interpreter) ของโปสต์สคริปต์มาใช้สร้างอุปกรณ์แสดงผล หรือเครื่องพิมพ์ ในแบบของตนเอง ที่ใช้งานร่วมกับคำสั่งภาษาโปสต์สคริปต์ได้

2) 1994 : PostScript level 2

ในปี 1994 Adobe ได้เปิดตัว PostScript level 2 ซึ่งมีการแก้ไขปรับปรุงส่วนที่สำคัญ ตามความต้องการของผู้ใช้ โดยคุณลักษณะที่สำคัญสำหรับ PostScript level 2 ได้แก่ เพิ่มความเร็วในการพิมพ์เป็น 4-5 เท่า โดยใช้เทคโนโลยีการสร้างภาพตัวอักษร (Font rendering) สนับสนุนการแยกสีในขั้นตอนของ RIP โดยรับเพิ่มโปสต์สคริปต์เข้าไปแยกสีในตัวเครื่องพิมพ์ หรือเครื่อง Imagesetter แทนที่จะต้องให้ซอฟต์แวร์บนเครื่องคอมพิวเตอร์ที่สั่งพิมพ์ เป็นผู้แยกสีออกมาเป็น CMYKให้ก่อน มีการย่อ (Compress) ข้อมูลก่อนส่ง เช่น ภาพกราฟิกชนิดบิตแมพ เมื่อเครื่องพิมพ์รับข้อมูลแล้วจึงจะไปขยายออกทำให้การส่งข้อมูลไปพิมพ์เร็วยิ่งขึ้น

สนับสนุนให้ใช้ฟอนต์ได้หลากหลาย โดยเฉพาะประเภทในแถบเอเชียที่มีตัวอักษรหลายรูปแบบมากกว่าในแถบยุโรป มีการพัฒนาไดรเวอร์เพิ่มมากขึ้น เช่น ไดรเวอร์ของ LaserWriter 8 ที่ทำงานกับเครื่อง Macintosh และไดรเวอร์ของ AdobePostScript เวอร์ชัน 2.X ที่ทำงานกับ Windows 3.1 มีการพัฒนาอัลกอริทึมในการแสดงผลบนหน้าจอ และการควบคุมกลไกการทำงานของเครื่องพิมพ์ได้ดีขึ้น รวมถึงการเลือกถาดป้อนกระดาษ ขนาดกระดาษด้วย PostScript Compatible ถึงแม้ PostScript level 2 จะเป็นที่นิยมใช้กันทั่วไป แต่ Adobe ก็ได้สร้างปัญหาใหญ่ขึ้น จากการตีพิมพ์สเปคของ level 2 ออกมาซึ่งมีการนำไปใช้งานจริง คือมีคู่แข่งที่ทำเลียนแบบ PostScript level 2 ขึ้นมา และทำงานได้เร็วกว่าของ Adobe ทำให้เกิดตลาดที่เรียกว่าเป็น PostScript Compatible คือซอฟต์แวร์ที่ใช้แทน PostScript interpreter ได้ทั้งในระดับของเครื่องพิมพ์เลเซอร์และเครื่องอิงก์แจตหรือ Imagesetter จนกลายเป็นคู่แข่งสำคัญและเป็นทางเลือกของผู้ที่ไม่อยากเสียค่าลิขสิทธิ์โปสต์สคริปต์ให้กับ Adobe

3) 1988 : PostScript 3

เพื่อแก้ปัญหาการสับสนในการเรียกชื่อ จึงใช้ว่า "PostScript 3" แทนที่จะเรียกเป็น "PostScript level 3" โดยถ้าเปรียบเทียบกับระหว่าง PostScript level 2 และ PostScript 3 แล้วมองดูเหมือนว่า PostScript 3 จะไม่ค่อยมีการเปลี่ยนแปลงอะไรที่สำคัญๆ เหตุผลอาจเป็นเพราะว่า แอปพลิเคชันจำนวนมากมีการใช้ PostScript level 2 กันแพร่หลายอยู่แล้ว แต่ข้อได้เปรียบของ PostScript 3 ได้แก่ สนับสนุนการไล่สีโทนสีเทา (Grayscale) ได้ถึง 4096 ระดับ ซึ่งถ้าเปรียบเทียบกับในอดีต จะมีการจำกัดระดับสีเทาไม่เกิน 256 ระดับเท่านั้น

เพิ่มจำนวนฟอนต์หลักของภาษาเป็น 136 ฟอนต์ สนับสนุนในการใช้ไฟล์ PDF (Portable Document Format) ที่สร้างจากโปรแกรม Acrobat ของ Adobe เองได้โดยตรง กล่าวคือ RIP ของ PostScript 3 พิมพ์ได้ทั้งเพิ่ม PostScript level 2 และเพิ่มพีดีเอฟเพิ่มความสามารถด้านกราฟิก ได้แก่ การผสมภาพบริเวณรอยต่อให้ดูกลมกลืนกัน รวมถึงการรองรับภาพแบบ 3 มิติด้วย ในการพิมพ์ออกเว็บ ได้มีการเพิ่มฟังก์ชันในการใช้งานอินเทอร์เน็ตเข้าไปในโพสต์สคริปต์ด้วย

2.2 Algorithm for comparing two different Printouts of the same PDF Document [3]

ในรูปแบบของความต้องการที่จะนำข้อมูลของแฟ้มพีดีเอฟมาเป็นรูปภาพนั้น บางครั้งอาจจะนึกถึงว่าการแปลงนั้นจะทำให้มีขนาดแฟ้มข้อมูลลดลง และทำให้คุณภาพของรูปที่ได้ นั้นอาจมีความคิดเห็นไปจากต้นฉบับมาก เนื่องมาจากการแปลงแฟ้มมีอัลกอริทึม (Algorithm) ที่ไม่ดีพอ อาจเกิดจากความต้องการหลาย ๆ ส่วนเช่นต้องการความเร็วในการแปลงข้อมูล หรือไม่มีความซับซ้อนในการถอดรหัสของแฟ้มพีดีเอฟ งานวิจัยของ Vaibhav Goel, ได้เสนออัลกอริทึมอีกแนวทางหนึ่งในการแปลงข้อมูลพีดีเอฟเป็นรูปแบบสกุลบีเอ็มพี (Bitmap Image) ซึ่งได้กล่าวถึงการแปลงข้อมูลโดยแยกออกเป็น 2 แบบ แล้วนำมาเปรียบเทียบกันเพื่อหาความ คิดเห็นของรูปที่ได้

2.3 PDF to HTML Conversion [4]

ในปัจจุบันคอมพิวเตอร์เป็นสิ่งที่มีผู้นิยมมาใช้ในการทำงานค่อนข้างมากตามแต่บุคคลใครจะใช้งานลักษณะใดบ้าง

ในเรื่องของงานเอกสารต่าง ๆ ที่มีใช้กันอยู่มากในขณะนี้ได้ทำเป็นเอกสารอิเล็กทรอนิกส์และต้องการให้สามารถอ่านเอกสารนั้นได้เพียงอย่างเดียว ซึ่งก็คือแฟ้มเอกสารพีดีเอฟ จากที่ได้อ่านมาแล้วการทำให้เอกสารพีดีเอฟจะทำมาจากแฟ้มเอกสารประเภทอื่น และเนื่องจากการทำเอกสารนั้นอาจจะมีการทำเอกสารที่มีการจัดวางของแถวข้อมูลได้ไม่ดัดนัก ทำให้อ่านได้ยาก จากงานวิจัยของ M. Afzal Bhatti, Adeel Ahmad, ได้มีแนวคิดที่จะนำแฟ้มเอกสารพีดีเอฟมาแปลงให้อยู่ในรูปแบบแฟ้มเอชทีเอ็มแอล (HTML) ซึ่งจะมีลักษณะของเนื้อหาแบบแถวเดียว ซึ่งทำให้อ่านได้ง่ายโดยอ่านจากบนลงล่าง โดยได้ใช้หลักการนำตัวอักษรจากแฟ้มพีดีเอฟมาจัดเรียงใหม่ในแฟ้มเอชทีเอ็มแอล

2.4 A New Method of Information Extraction from PDF Files [5]

การเปิดเอกสารในแฟ้มพีดีเอฟที่มีจำนวนหน้าปริมาณมาก ทำให้ยากต่อการที่จะค้นหาหัวข้อ ชื่อเรื่องในส่วนต่าง ๆ ภายในเอกสาร ซึ่งในแฟ้มพีดีเอฟหลายแฟ้มที่ไม่ได้ทำดัชนี (Index) หัวข้อของเนื้อหา โดย FANG YUAN และ BO LIU ได้มีแนวคิดในการทำวิจัย เพื่อแก้ปัญหาที่นี้จะสามารถทำการค้นหา และดึงข้อมูลในส่วนที่เป็นหัวข้อหลักออกมาให้ผู้ใช้ได้รับรู้ โดยได้เสนอหลักการคือใช้การสร้างกฎ (Rule) ในการดึงหัวข้อ และคำค้นหลัก ๆ (Keyword) แล้วทำการเริ่มต้นข้อมูลที่อยู๋ภายในแฟ้มพีดีเอฟด้วย "Tree-like model" ในขั้นตอนนี้จะทำการสแกนข้อมูลไปทีละส่วน เมื่อค้นพบคำหรือส่วนที่อยู๋ในกฎ และคำค้น ก็จะทำการระบุ (Tag) ข้อมูลจุดนั้นไว้ แต่การที่จะให้ระบบการค้นหาให้ได้ตามกฎนั้นจะต้องอาศัยหลักการในการเรียนรู้ชุดข้อมูลต่าง ๆ ว่าลักษณะของข้อมูลเป็นรูปแบบใด ทั้งนี้มีผลสรุปของการดึงข้อมูลได้ของระบบคือ นำเอกสารสำหรับให้ระบบทำการเรียนรู้จำนวน 40 ฉบับ ให้เอกสารสำหรับการทดสอบจำนวน 200 ฉบับ ทำการค้นหาได้จำนวน 179 ฉบับ

2.5 ImageMagick [6]

"ImageMagick" คือโปรแกรมเปิด (Open Source) ที่พัฒนาโดย ImageMagick Studio LLC, โดยมีความสามารถสร้าง แก้ไข ด้วยการใช้การประกอบรูปภาพจากจุดหลายๆจุดรวมกัน และโปรแกรมยังสามารถอ่าน แปลง เขียน รูปภาพได้หลาย

ชนิด สามารถตัดรูปภาพเฉพาะส่วนที่ต้องการ เปลี่ยนสี แสดงผลรูปภาพหลายแบบ (Effects) และเนื่องจากเป็น โปรแกรมฟรีจึงสามารถที่จะนำรหัสต้นฉบับ (Source Code) มาใช้ คัดแปลงแก้ไข และ พัฒนาใหม่ได้ และไม่จำกัดเรื่องของระบบปฏิบัติการที่ใช้ จึงสามารถทำงาน ได้บนเครื่องแม่ข่าย

“ImageMagick” รองรับเพิ่มรูปภาพได้กว่า 90 ชนิด (GIF, JPEG, JPEG-2000, PNG, PDF, PhotoCD, TIFF, and DPX.)

3. แนวคิดและวิธีการดำเนินการวิจัย

จากการศึกษาริธีการนำข้อมูลจากแฟ้มพีดีเอฟออกมาโดยใช้วิธีการ ใช้เพิ่มรูปภาพแสดงแทนสามารถแบ่งเป็นขั้นตอนการดำเนินงานตามการใช้งานของผู้ใช้ และขั้นตอนการทำงานของฟังก์ชันที่จะนำมาใช้เพิ่มเติมคือ การกรองข้อมูลภาพออกจากแฟ้มพีดีเอฟ การค้นหาค่าในแฟ้มพีดีเอฟ การดึงชื่อเรื่องหลักจากเอกสาร

3.1 ภาพรวมของการแสดงผลเอกสาร

เมื่อผู้ใช้ทำการเริ่มใช้บริการระบบจะ ทำการนำแฟ้มพีดีเอฟที่ต้องการดู มาอ่านข้อมูลภายในทั้งหมดและจะสามารถรู้ได้ว่าแฟ้มพีดีเอฟนี้มีจำนวนทั้งหมดกี่หน้า จากนั้นจะส่งข้อมูลเพียง 2 หน้าแรกของเอกสาร ไปยังส่วนของการแปลงแฟ้มพีดีเอฟเป็นแฟ้มเจพีก เมื่อได้เพิ่มรูปภาพชนิดเจพีกแล้ว จะถูกนำไปแสดงผลบนเครื่องของผู้ใช้บนเว็บเบราว์เซอร์ โดยมีขั้นตอนการทำงานแสดงดังภาพที่ 1



ภาพที่ 1: ภาพรวมของการแสดงผลเอกสาร

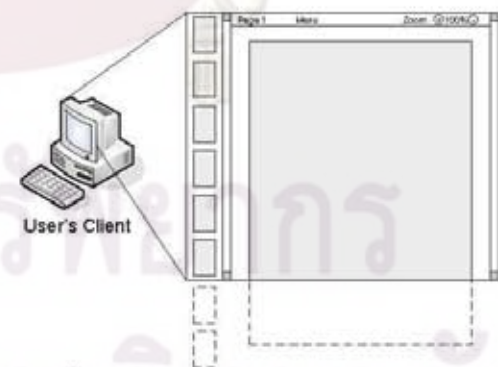
จะเห็นได้ว่าการแปลงข้อมูลแฟ้มพีดีเอฟเป็น โฟลต์ สกริปต์ แล้วนำไปสร้างเป็นภาพ (Bitmap Image) จากนั้น จะทำการบีบอัดข้อมูลให้เป็นแฟ้มเจพีก จะใช้เอพีไอ (API - Application Programming Interface) ชื่อ “ImageMagick” เนื่องจาก ในกระบวนการหรือโปรแกรมที่ใช้สร้างแฟ้มพีดีเอฟนั้นมีมาก และใช้มาตรฐานที่แตกต่างกัน อีกทั้งประเภทของอักษรที่ใช้ในปัจจุบันมีจำนวนมาก ด้วยเหตุนี้จึงจำเป็นที่จะต้องนำเอพีไอเข้ามาช่วยในการแปลงแฟ้มเอกสาร และสามารถเลือกใช้ “Imagemagick” ก็เพราะเป็น Open Source ที่สามารถนำรหัสต้นฉบับ (Source code) มาใช้พัฒนาได้

3.2 ขีดความสามารถในส่วนการแสดงผลบนหน้าเว็บ

สามารถแบ่งเป็นขีดความสามารถต่าง ๆ ได้ดังต่อไปนี้

3.2.1 ขีดความสามารถในการนำแฟ้มเจพีกแสดงผลบนเว็บเบราว์เซอร์

การออกแบบยูไอ (UI – User Interface) โดยจะมีการนำเอารูปเจพีกที่ได้จากการแปลงเอกสารมาวางในตำแหน่งของแต่ละหน้า ซึ่งจะมีการเตรียมพื้นที่สำหรับให้หน้าเอาเพิ่มรูปภาพจัดวางลงไป และสามารถทำการเลื่อนเอกสารได้โดยการคลิกหรือคลิก (Click) ปุ่มแล้วขยับขึ้นหรือลงได้ จะเห็นได้ว่ามีการแบ่งออกเป็นสองส่วนคือ ส่วนที่ใช้ดูเอกสารโดยรวม (ช่องทางด้านซ้าย) และส่วนแสดงเอกสารหลัก (ช่องทางด้านขวา) ดังภาพที่ 2 ซึ่งในช่องนี้ส่วนบนจะมีเครื่องมือในการใช้งานฟังก์ชันตามที่มีในงานวิจัยนี้



ภาพที่ 2: User Interface ของหน้าแสดงผลเอกสารบนเบราว์เซอร์



ภาพที่ 3: วิธีการวางชั้นรูปภาพเพื่อแสดงผล

จากภาพที่ 3 จะเห็นได้ว่า การแสดงรูปนั้นจะมีการจัดการแบบระดับชั้น โดยมีรูปของเอกสารวางในชั้นแรก แล้วชั้นบนสุดคือชั้นสำหรับรับพฤติกรรมของผู้ใช้ เช่น การจับวัตถุในหน้าแล้วเคลื่อนเอกสารไปในทิศทาง ขึ้น ลง ขวา ซ้าย และจากฟังก์ชันการค้นหา จะมีการนำเพิ่มรูปอีกชนิดหนึ่งคือเพิ่มจ๊อบที่มีแถบสีระบาศไว้เป็นจุด ๆ มาแทรกระหว่างชั้นของรูปเอกสารกับชั้นรับพฤติกรรมผู้ใช้ ทำให้มีการซ้อนกันของรูปภาพ ซึ่งจะแสดงให้เห็นว่าคำที่ทำการค้นหาคือตำแหน่งใด ในส่วนการทำงานนี้จะกล่าวในขั้นตอนต่อไป

ในการออกแบบแล้วพัฒนาไอนี้จะใช้ความรู้ในเรื่องของชีเอสเอส (Cascading Style Sheets) [7] และวาจาสคริปต์ [8]

3.2.2 ขีดความสามารถในการควบคุมการแปลงเพิ่มพีดีเอฟ

จากขั้นตอนข้างต้นเมื่อผู้ใช้ต้องการเลือกโดยการคลิกเลือกหน้าที่ช่องทางซ้าย หรือเปลี่ยนหน้าเอกสาร โดยการเคลื่อนหน้าเอกสารขึ้นลง ระบบจะทำการตรวจสอบความต้องการนั้นจากชั้นรับพฤติกรรมผู้ใช้ แล้วส่งคำสั่งความต้องการเพิ่มรูปภาพเอกสารหน้าที่ผู้ใช้งานต้องการเลือก โดยใช้เอแจ็ทส์ส่งคำสั่งนั้นกลับไปยังเครื่องแม่ข่าย

ระบบจะรับคำสั่งนั้นมาและสั่งให้การแปลงเพิ่มพีดีเอฟเป็นเพิ่มรูปภาพในหน้าที่ต้องการ เพื่อส่งกลับไปยังเครื่องของผู้ใช้อีกครั้ง ซึ่งขั้นตอนนี้จะคล้ายกับขั้นตอนแรก เพียงแต่มีการระบุหน้าของเอกสารที่ต้องการ แต่ถ้าผู้ใช้กลับไปเลือกหน้าที่ได้เคยเปิดไว้แล้วระบบจะทำการตรวจสอบว่าได้สร้าง

หน้าไว้แล้วก็จะไม่แปลงเพิ่มรูปในหน้านั้นขึ้นมาอีก การสร้างเอกสารเป็นเพิ่มเจพีคแต่ละรูปจะมีการเก็บไว้ในที่เก็บชั่วคราว เมื่อถึงระหว่างเวลาที่ไม่มีการใช้งาน (Time out) ระบบจะทำการลบเพิ่มนั้นออกไป

3.2.3 ขีดความสามารถในการย่อขยายเอกสาร

เมื่อระบบรับข้อมูลให้มีการย่อขยาย ระบบจะส่งงานตรงไปยังส่วนการสร้างเพิ่มเจพีคที่อยู่ในระบบของ "ImageMagick" โดยจะมีการระบุขนาดของเอกสารในแนวกว้าง โดยใช้ความละเอียดอยู่ที่ 72 DPI (Dot Per Inch) ซึ่งจะมีด้านกว้างเท่ากับ 595 จุด (Pixel) เทียบเท่ากับ 100% ของการแสดงผลเอกสาร เมื่อมีการสั่งให้ย่อขยายในช่วง 10 – 800% ให้เทียบสัดส่วนโดยนำ 5.95 คูณกับขนาดเปอร์เซ็นต์ที่ต้องการ จะได้ขนาดความกว้างเป็นจุด

ความสูงให้เทียบสัดส่วนของความกว้างความสูงของเอกสารนั้น ซึ่งจะมีความกว้างเท่ากัน

3.2.4 ขีดความสามารถในการค้นหาในแฟ้มพีดีเอฟ

การค้นหาคำในกรณีนี้เป็นเอกสารที่เป็นภาษาไทยในแฟ้มพีดีเอฟ จะใช้วิธีการค้นหาโดยวิธีการเปรียบเทียบสายอักขระตามแนวคิดของ Brute-force [9] คือ ทำการเปรียบเทียบโดยเลื่อนอักขระที่ทำการเปรียบเทียบไปทางขวามือครั้งละ 1 อักขระ การค้นข้อความไทยในแฟ้มพีดีเอฟเลือกใช้วิธีนี้เนื่องจาก กลุ่มอักขระที่ใช้ในการเปรียบเทียบทั้ง 2 กลุ่มมีขนาดไม่ยาวมาก ไม่ว่าจะเลือกวิธีการเปรียบเทียบแบบใดก็ใช้เวลาในการค้นข้อความไม่แตกต่างกัน แต่การค้นหาข้อความไทยในแฟ้มพีดีเอฟ ใช้การเปรียบเทียบตามวิธีการของ Brute-force [9] อย่างเดียวไม่เพียงพอ ต้องมีการดัดแปลงการทำงานบางอย่างเพิ่มเติมเข้าไปด้วย ทั้งนี้เนื่องจากผลลัพธ์ที่ได้จากการถอดรหัสข้อความไทยเป็นปัญหาและอุปสรรคในการค้นข้อความ

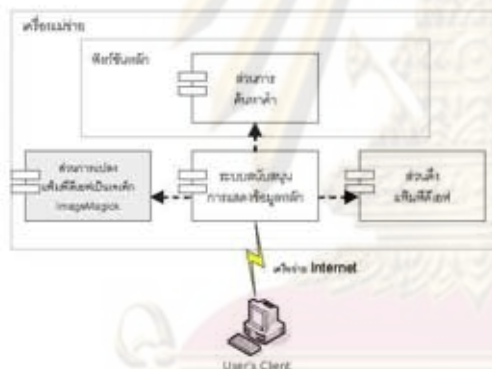
การเข้ารหัสข้อความภาษาไทยตามข้อกำหนด มอก. 620 ผลลัพธ์ของกระบวนการนี้จะได้กลุ่มของอักขระครั้งละหนึ่งกลุ่ม ซึ่งข้อความภาษาไทยจะมีการแบ่งได้หลายกลุ่ม

เมื่อกระบวนการค้นหาทั้งหมดทำการค้นหาอยู่นั้นจะมีการตรวจสอบตำแหน่งของอักขระนั้นด้วยและเก็บข้อมูลนั้นมาแสดงเป็นรายการตำแหน่งของคำที่ค้นหา และทำการ

สร้างเพิ่มโพสต์สคริปต์ซึ่งจะมีเพียงแถบสีวางอยู่ในตำแหน่งของอักขระที่ค้นหาได้ โดยในขั้นตอนนี้เมื่อได้เพิ่มโพสต์สคริปต์แล้วจะถูกส่งไปขั้นตอนการสร้างเพิ่มรูป แต่จะมีการเปลี่ยนให้เป็นเพิ่มรูปภาพชนิดจีฟ (GIF - Graphics Interchange Format) โดยจะนำไปวางทับกับรูปเอกสารหลัก ทำให้เห็นถึงตำแหน่งคำที่ทำการค้นหา

4. การพัฒนาระบบ

งานวิจัยนี้ได้ทำการพัฒนาและออกแบบสถาปัตยกรรมของระบบ โดยมีระบบสนับสนุนการแสดงผลข้อมูล เป็นตัวกลางในการควบคุมการทำงาน และฟังก์ชันทั้งหมดที่อยู่ในระบบตามที่ได้กล่าวมาข้างต้น จะเห็นได้ว่าฟังก์ชันหลักนั้นจะขึ้นตรงกับระบบสนับสนุนการแสดงผลข้อมูล ซึ่งจะรับคำสั่งการทำงานโดยตรงจากเครื่องผู้ใช้ ดังภาพที่ 4



ภาพที่ 4: สถาปัตยกรรมของระบบสนับสนุนโปรแกรมค้นหาเว็บพีดีเอฟแบบออนไลน์ โดยใช้เพิ่มรูปภาพ

5. บทสรุป

งานวิจัยนี้ต้องการที่จะนำเสนอการอ่านข้อมูลภายในเพิ่มเอกสารอิเล็กทรอนิกส์ประเภทพีดีเอฟได้โดยไม่ต้องมีโปรแกรมประยุกต์ที่ใช้สำหรับเปิดเพิ่มพีดีเอฟนี้ และยังมีความสามารถในการอ่านข้อมูลได้ทันทีไม่ต้องดาวน์โหลดเอกสารนั้นมาก่อนทั้งหมด จึงทำให้สามารถที่จะอ่านเอกสารพีดีเอฟได้โดยไม่ต้องคำนึงถึงขนาดความจุของเพิ่มเอกสารนั้นๆ ซึ่งในงานวิจัยนี้ได้รวมเทคโนโลยีที่ใช้กันบนเว็บไซต์และงานวิจัยเกี่ยวกับการเข้าถึงข้อมูลภายในเพิ่มเอกสารพีดีเอฟ ใที่มีลักษณะการทำงานและฟังก์ชันในการใช้งาน

คล้ายคลึงกับ โปรแกรมประยุกต์ที่ใช้กันอยู่ในปัจจุบัน และยังสามารถนำไปใช้ร่วมกับเว็บไซต์เกี่ยวกับการนำเสนอ บทความ งานวิจัย วิทยานิพนธ์ ซึ่งเก็บเอกสาร ในลักษณะเพิ่มพีดีเอฟ และมีจุดประสงค์ที่จะเผยแพร่ข้อมูลเพียงบางส่วนนั้น ก็ทำได้โดยไม่ต้องทำการตัดต่อหรือตัดแปลงเพิ่มต้นฉบับ เพราะการแสดงข้อมูลนั้นเป็นการแสดงโดยใช้เพิ่มรูปภาพแสดงเอกสารนั้นแทนโดยแบ่งเป็นหน้าต่อหน้ารูป จึงทำให้จำกัดหน้าที่จะสามารถทำได้โดยง่าย

เอกสารอ้างอิง

- [1] Adobe System Incorporated. Portable Document Format Reference Manual, Adobe Systems 1999; P 19-21.
- [2] Adobe PostScript 3. [Online]. Available from: <http://www.adobe.com/products/postscript/> [March, 2008]
- [3] Vaibhav Goel. Algorithm for comparing two different Printouts of the same PDF Document, Department of Computer Engineering, Netaji Subhash Institute of Technology, University of Delhi, India, 2005.
- [4] M. Afzal Bhatti, Adeel Ahmad. PDF to HTML Conversion; Having a Usable Web Document. Department of Computer Science, Quaid-i-Azam University, Islamabad, Pakistan, 2006.
- [5] FANG YUAN, BO LIU. A New Method of Information Extraction from PDF Files. Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005.
- [6] ImageMagick. [Online]. Available from: <http://www.imagemagick.org/script/index.php> [March, 2008]
- [7] Javascript.[Online]. Available from: <http://www.w3schools.com/js/default.asp> [March, 2008]
- [8] Cascading Style Sheets. [Online]. Available from: <http://www.w3.org/Style/CSS/> [March, 2008]
- [9] Brute Force algorithm. [Online]. Available from: <http://igm.univ-mlv.fr/~lecroq/string/node3.html> [January, 1997]

ประวัติผู้เขียนวิทยานิพนธ์

นายอัศวิน วงษ์แก่นคำ เกิดเมื่อวันที่ 16 ตุลาคม พ.ศ. 2522 ที่จังหวัดอุบลราชธานี สำเร็จ การศึกษาหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชาวิศวกรรมโยธา จากภาควิชาวิศวกรรม โยธา คณะครุศาสตร์อุตสาหกรรม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ในปีการศึกษา 2544 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ที่ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2549



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย