การสร้างระบบเพื่อใช้ลายเซ็นดิจิตอลในการลงนามร่วมกัน

นาย วิชัย ทศมาศวรกุล

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2543

ISBN 974-346-958-3

SYSTEM DEVELOPMENT FOR MULTI SIGNATORY USING DIGITAL SIGNATURES

Mr. Wichai Tossamartvorakul

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2000

ISBN 974-346-958-3

หัวข้อวิทยานิพนธ์	การสร้างระบบเพื่อใช้ลายเซ็นดิจิตอลในการลงนามร่วมกัน
โดย	นายวิชัย ทศมาศวรกุล
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร.ยรรยง เต็งอำนวย

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร. สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ บุญชัย โสวรรณวณิชกุล)

..... อาจารย์ที่ปรึกษา
(อาจารย์ ดร.ยรรยง เต็งอำนวย)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. สมชาย ประสิทธิ์จูตระกูล)

..... กรรมการ
(คุณสุภัทรียา จิตรกร)

วิชัย ทศมาศวรกุล : การสร้างระบบเพื่อใช้ลายเซ็นดิจิตอลในการลงนามร่วมกัน (SYSTEM DEVELOPMENT FOR MULTI SIGNATORY USING DIGITAL SIGNATURES) อ. ที่ปรึกษา : ดร. ยรรยง เต็งอำนวย, หน้า 99 ISBN 974-346-958-3

ในปัจจุบันยังมีการใช้งานเอกสารกระดาษอยู่แทนการใช้งานคอมพิวเตอร์ เนื่องจากคอมพิวเตอร์ยังไม่สามารถแทนที่การทำงานของกระดาษได้ในบางด้านเช่น เอกสารที่ต้องมีการลงนามที่แตกต่างกัน เช่น ลงนามร่วมกัน ลงนามตามลำดับชั้น เป็นต้น

วิทยานิพนธ์ฉบับนี้มีจุดประสงค์ในการศึกษาและออกแบบโครงสร้างเอกสารที่สามารถรองรับ การลงนามร่วมกันมากกว่าหนึ่งคน ตลอดจนการตรวจสอบการแก้ไขของเอกสาร โดยเริ่มจากทำการศึกษาถึงความต้องการของเอกสารที่ต้องมีการลงนามในแบบต่างๆ นำมาออกแบบโครงสร้างเอกสาร โดยในวิทยานิพนธ์ฉบับนี้ใช้โครงสร้างเอกสารเอกซ์เอ็มแอล (XML) เป็นหลักในการออกแบบ รวมทั้งการทดลองสร้างระบบต้นแบบเพื่อทดสอบโครงสร้างเอกสารที่ออกแบบว่าสามารถนำไปใช้งานจริงได้ดีเพียงใด เพื่อนำไปสู่การปรับปรุงโครงสร้างและเพิ่มเติมแก้ไขต่อไปในอนาคต พร้อมทั้งเสนอแนวทางในการพัฒนาระบบให้สมบูรณ์ เพื่อนำไปใช้งานจริงได้กับโปรแกรมประยุกต์ต่างๆ ที่มีอยู่แล้ว

จากผลการวิจัยโครงสร้างเอกสารที่ออกแบบสามารถตอบสนองความต้องการหลักๆ ในส่วนของการลงนามร่วมกัน การกำหนดสิทธิ์ในการแก้ไขของเอกสาร และการตรวจสอบประวัติการแก้ไขเอกสาร แต่ยังมีส่วนต้องปรับปรุงเพิ่มเติมในเรื่องของการเข้ารหัสข้อมูลในเอกสาร และการป้องกันการแก้ไขตัวโครงสร้างเอกสารเอกซ์เอ็มแอล ซึ่งจะมีผลต่อการเปลี่ยนคุณสมบัติของเอกสารในด้านผู้มีสิทธิ์ใช้และแก้ไขเอกสาร

ภาควิชา วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต .....
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์	ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2543	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

4071475221 : MAJOR COMPUTER SCIENCE
KEY WORD: DIGITAL SIGNATURES / SIGNED DOCUMENT / XML

WICHAI TOSSAMARTVORAKUL : SYSTEM DEVELOPMENT FOR MULTI SIGNATORY USING DIGITAL SIGNATURES. THESIS ADVISOR : YUNYONG TENG-AMNUAY,Ph.D, 99 pp.
ISBN 974-346-958-3

Paper documents are still being used instead of electronic documents. This is because computer cannot replace paper document in its entirety especially the document that needs signatory, i.e. co-signatory, hierarchical signatory.

This thesis focuses on studying and designing electronic document structure that supports co-signatory. The scope of the thesis also includes document authentication. The requirements for various types of signatory document are used as the base line for electronic document structure designing. XML is used in the design of document structure. The thesis includes the development of a prototype to verify the designed document. Suggestion for improvement is provided so that the improvement can be done in the future.

The document designed in this thesis supports the requirement in co-signatory, document security and document versioning. However, there is a room for improvement in data encryption and protecting alteration of XML document. So that document structure can be maintained.

Department of Computer Engineering	Student's signature .....
Field of study Computer Science	Advisor's signature .....……
Academic year 2000	Co-advisor's signature .....……

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของอาจารย์ ดร.ยรรยง เต็งอำนวย อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่าง ๆ ในการวิจัยด้วยดีตลอด และขอขอบคุณ คุณสุภัทรียา จิตรกรจากสถาบันวิทยบริการ ที่ให้ข้อมูลและตัวอย่างเอกสารที่ใช้ในงานราชการ ในขั้นตอนการวิเคราะห์ความต้องการของเอกสารที่ต้องมีการลงนาม

ท้ายนี้ผู้วิจัยใคร่ขอขอบคุณภรรยาและลูกสาวที่เป็นกำลังใจ และช่วยด้านการพิมพ์เอกสารแก่ ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สารบัญ

สารบัญตาราง

สารบัญรูปภาพ

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เอกสารที่สำคัญในปัจจุบันมักมีการลงนามเพื่อรับรองข้อความในเอกสาร ซึ่งการลงนามอาจมีได้หลายแบบ เช่น การลงนามร่วมกัน การลงนามตามสายงานบังคับบัญชา เช่น หัวหน้าแผนก ผู้จัดการแผนกจนถึงประธานบริษัท ในแต่ละขั้นตอนของการลงนามอาจมีการแก้ไข เพิ่มเติม ในเอกสาร หรือมีการแนบเอกสารเพื่อการพิจารณา เช่น การขออนุมัติซื้อสินค้าที่มีใบเสนอราคาแนบมาด้วย ซึ่งผู้อนุมัติอาจอนุมัติแต่ต้องมีการแก้ไขข้อมูลบางส่วน เช่น จำนวนสินค้าที่สั่ง

เอกสารบางฉบับเมื่อมีการลงนามแล้วไม่สามารถแก้ไขได้ เช่น การเซ็นชื่อในเช็ค หรือ เอกสารทางการเงินต่างๆ เอกสารที่เป็นกระดาษมักใช้ลายเซ็นในการบ่งชี้ถึงผู้ที่ได้รับอำนาจ โดยถือว่าผู้ที่ลงลายเซ็นได้อ่านและรับทราบข้อความแล้ว แต่อาจมีปัญหาเกิดขึ้นได้ในกรณีที่มี การแก้ไขเอกสารหลังจากเซ็นชื่อแล้ว เช่น การเพิ่มเติมข้อความ ซึ่งในกรณีนี้ยากแก่การตรวจสอบ ในทางปฏิบัติมักมีการใส่ตัวอักษรที่ไม่มีความหมาย เช่น * หรือ – ลงในช่องว่างเพื่อป้องกัน การเพิ่มเติมภายหลัง

ปัจจุบันเทคโนโลยีของเอกสารบนอินเทอร์เน็ตได้รับการยอมรับอย่างแพร่หลายในแง่ของการใช้งานในด้านการเผยแพร่ข่าวสาร เนื่องจากแก้ไขข้อมูลที่จุดศูนย์กลางเพียงอย่างเดียว คนอื่นสามารถเข้ามาใช้งานร่วมกัน จึงทำให้เอกสารได้รับการเก็บอยู่ในรูปแบบทางอินเทอร์เน็ตมากขึ้น เช่น รูปแบบของ เอชทีเอ็มแอล แต่เอกสารสำคัญที่มีการลงนามไม่สามารถจัดเก็บในรูปแบบของอินเทอร์เน็ตได้ เนื่องจากว่าโครงสร้างของเอกสารบนอินเทอร์เน็ตยังไม่มีฟังก์ชั่นการทำงานในส่วนที่รองรับในเรื่องของการลงนามในเอกสารทั้งแบบคนเดียว ร่วมกันหรือตามลำดับขั้น ทำให้งานบนอินเทอร์เน็ตถูกจำกัดอยู่ในงานที่สามารถเปิดเผยข้อมูลได้ทั่วไป และไม่จำเป็นต้องมีการตรวจสอบความถูกต้อง ไม่สามารถใช้กับงานเอกสารที่ต้องมีการลงนาม เช่น เอกสารทางการเงินได้ โครงสร้างของเอกสารที่นิยมใช้กันในปัจจุบันคือ เอชทีเอ็มแอล (Hyper Text Mark Up Language) แต่ เอชทีเอ็มแอล ออกแบบมาใช้ในงานนำเสนอ (Presentation) ของข้อมูลทางเว็บเพจ (Web Page) เท่านั้น จากการที่เอชทีเอ็มแอล ถูกออกแบบมาเพื่อใช้ในงานการแสดงข้อมูล เช่น แสดงตัวอักษรหนา เอียง ขนาด ตำแหน่งตาราง ทำให้มีข้อจำกัดหลายประการ คือ

1. มีโครงสร้างที่จำกัด ซึ่งใช้ในการจัดรูปแบบเอกสารเท่านั้น ไม่สามารถแสดงลักษณะ ของข้อมูลหรือความหมายของข้อมูลในเอกสารได้
2. การแก้ไขเพิ่มเติมข้อความในเอกสารต้องมีการจัดรูปแบบของเอชทีเอ็มแอลในเอกสารใหม่ เช่น เติมหัวข้อย่อยเข้าไป ในเอกสารต้องเลื่อนหัวข้ออื่นที่อยู่ภายใต้หัวข้อย่อยเข้าไปอีกขั้นหนึ่ง

3. การแลกเปลี่ยนข้อมูลระหว่างเอกสาร เอชทีเอ็มแอล ทำได้ยาก เช่น ต้องการย้ายชื่อของรถจากเอชทีเอ็มแอล หนึ่งไปอีก แฟ้มข้อมูลหนึ่งไม่สามารถทำได้ เพราะเอกสาร เอชทีเอ็มแอล ไม่สามารถบอกคุณลักษณะของข้อมูลได้ว่า ส่วนไหนเป็นชื่อรถ

4. การค้นหาข้อความในเอกสาร เอชทีเอ็มแอล มักได้ข้อความที่ไม่เกี่ยวข้องด้วย เช่น การคำว่า “เขียว” ซึ่งหมายถึงสี อาจได้คำตอบหลายคำตอบที่ไม่เกี่ยวข้อง เช่น คนชื่อ “เขียว” หรือ อาการ “หน้าเขียว” (โกรธ) เนื่องจากเอกสาร เอชทีเอ็มแอล ไม่สามารถเข้าใจและแยกความหมายของคำว่า “เขียว” ในเอกสารได้

จากปัญหาข้างต้นจึงได้มีการคิดค้นภาษาแบบมาร์คอัพ (Markup Language) ใหม่ซึ่งสามารถบอกความหมายของเนื้อความในเอกสารได้ว่าหมายถึงอะไร ซึ่งรูปแบบของเอกสารในปัจจุบันที่เริ่มได้รับความนิยมกันคือ XML (Extensible Markup Language)

ข้อดีของเอกสาร XML คือ

1. โครงสร้างของ XML สามารถเพิ่มเติมได้ กล่าวคือ สามารถสร้าง Markup Language เพื่อใช้งานในด้านต่างๆได้

2. XML สามารถแสดงถึงความหมายของข้อมูลได้ทำให้สามารถนำข้อมูลไปใช้งานที่ต้องเกี่ยวกับการประมวลผลได้ เช่น การค้นหาคำที่ได้ความหมายตามที่ต้องการ การเชื่อมโยงของข้อมูลประเภทเดียวกัน ตัวอย่างเช่น

```
<Book>
<author> John </author>
<Title> New World </Title>
</Book>
```

จากตัวอย่างข้างบนจะเห็นได้ว่าโปรแกรมสามารถทราบได้ว่าผู้แต่งหนังสือ New World คือใคร โดยไม่ต้องไปค้นหาความสัมพันธ์ของข้อมูลในฐานข้อมูล

3. มีโปรแกรมสนับสนุน เช่น IE 4.0 ของไมโครซอฟท์ และโปรแกรมอื่นๆ ที่ประกาศจะสนับสนุน XML

จากสาเหตุที่เอกสารบนอินเทอร์เน็ตได้รับการใช้งานอย่างแพร่หลาย แต่ยังไม่สามารถใช้งาน กับเอกสารที่ต้องมีการลงนามรับรอง ทำให้เกิดแนวคิดของวิทยานิพนธ์ฉบับนี้ที่จะค้นหาวิธีในการกำหนดและ ออกแบบรูปแบบเอกสารอิเล็กทรอนิกส์ที่สามารถรองรับการใช้งานในด้านการลงนามเอกสารอิเล็กทรอนิกส์ ซึ่งในวิทยานิพนธ์ฉบับนี้มุ่งเน้นไปในรูปแบบเอกสารของ XML ในการพัฒนาต้นแบบ เนื่องจากมีความยืดหยุ่นและพัฒนาได้ดีกว่าเอชทีเอ็มแอลซึ่งออกแบบเพื่อใช้ในการแสดงผลเพียงอย่างเดียว

การเปลี่ยนรูปแบบการลงนามโดยกระดาษมาเป็นการลงนามบนเอกสารทางอินเทอร์เน็ตทำให้เกิดปัญหาที่ต้องได้รับการศึกษาและแก้ไขดังต่อไปนี้

1. ทำอย่างไรเมื่อมีคนลงนามร่วมกันมากกว่า 1 คนในกรณีที่เป็นการเซ็นร่วมกัน และเซ็นตามลำดับสายงาน เช่น หัวหน้าแผนกไปยังผู้จัดการ

2. ทราบได้อย่างไรว่าคนที่เซ็นชื่ออนุมัตินั้นเป็นบุคคลนั้นจริง ไม่ใช่ผู้แอบอ้าง เนื่องจากเป็นการกระทำทางอิเล็กทรอนิกส์ ซึ่งข้อมูลสามารถปลอมแปลงได้ง่ายกว่า การเซ็นชื่อลงบนกระดาษ

3. ไม่มีการเปลี่ยนแปลงเอกสารหลังจากมีการอนุมัติแล้ว ทำอย่างไรจึงจะแน่ใจได้ว่าเอกสารไม่มีการแก้ไข ต่อเติม เมื่อมีการเซ็นชื่อทางอิเล็กทรอนิกส์แล้ว

4. ในกรณีที่มีผู้อนุมัติหลายคน เอกสารต้องมีการผ่านมากกว่าหนึ่งคนขึ้นไป และคนที่อนุมัติถัดมาอาจมีความสามารถในการแก้ไขเอกสารได้ เช่น ผู้จัดการอาจเปลี่ยนแปลงเงื่อนไขบางข้อหลังจากได้รับเอกสารจากหัวหน้าแผนก ถ้าเป็นเอกสารทาง กระดาษสามารถทราบได้จากรอยขีดหรือลบบนกระดาษ ซึ่งหัวหน้าแผนกจะรับผิดชอบ ในส่วนที่ตัวเองแก้ไขเท่านั้น ผู้จัดการต้องรับผิดชอบเพิ่มเติมในส่วนที่ตัวเองแก้ไขเพิ่มเติม เมื่อนำระบบคอมพิวเตอร์มาใช้จึงเกิดปัญหาว่าจะตรวจสอบได้อย่างไรว่าผู้เซ็นชื่อก่อนรับผิดชอบเฉพาะส่วนที่ตัวเองแก้ไขเพิ่มเติม ไม่ต้องรับผิดชอบในส่วนที่ผู้อนุมัติคนอื่นเข้าไปแก้ไขเพิ่มเติม เนื่องจากผู้อนุมัตินั้น อาจจะไม่เห็นการแก้ไขเอกสารซึ่งตนเองอนุมัติผ่านไปแล้วโดยบุคคลที่มีอำนาจสูงกว่า

5. การแนบเอกสารประกอบเมื่อมีการเซ็น และเอกสารประกอบต้องไม่สามารถแก้ไขหรือ เปลี่ยนแปลง เมื่อมีการเซ็นเกิดขึ้น

6. ต้องการดูเอกสารฉบับก่อนที่มีการเซ็นและแก้ไขโดยแต่ละคน หรือส่วนที่มีการแก้ไขของแต่ละคน

1.2 วัตถุประสงค์

1. ศึกษาขั้นตอนและขบวนการการทำงานร่วมกันหลายคนของเอกสารราชการ เอกสารทางการเงินที่มีการลงนามโดยใช้กระดาษในรูปแบบต่างๆ เช่น การลงนามร่วมกัน การลงนามตามลำดับขั้น

2. ศึกษาและออกแบบรูปแบบของเอกสารอิเล็กทรอนิกส์ที่สามารถตรวจสอบตัวจริงของเอกสารและการลงนามทางอิเล็กทรอนิกส์

3. สามารถนำโครงสร้างเอกสารที่ออกแบบไปใช้กับรูปแบบเอกสารอื่นๆได้ แต่ในการทำวิทยานิพนธ์จะยึด XML เป็นหลัก

1.3 ขอบเขตการวิจัย

1) ศึกษาถึงรูปแบบของการลงนามเฉพาะการลงนามคนเดียว การลงนามร่วมกันและการลงนามตามสายบังคับบัญชาของเอกสารทางราชการโดยมุ่งเน้นเอกสารทางการเงิน

2) ออกแบบโครงสร้างเอกสารที่รองรับการลงนามร่วมกัน ในลักษณะของ Markup Language

3) กำหนดความหมาย คุณลักษณะ (SPECIFICATION) ของแต่ละแทก(TAG) ใน Markup Language ว่ามีการทำงานอย่างไร

4) ออกแบบในส่วนการเก็บข้อมูลของเอกสารทางอิเล็กทรอนิกส์ที่มีการลงนามในรูปของแฟ้มข้อมูล

a. ในส่วนของโครงสร้างหลักของเอกสารที่ออกแบบต้องมีความสามารถหลักดังต่อไปนี้

i. ความสามารถในการพิสูจน์ตัวจริงของเอกสาร (Authenticate)

ii. การลงลายเซ็นอิเล็กทรอนิกส์ในแต่ละคน ทั้งแบบ คนเดียว สองคนร่วมกันและตามลำดับสายงาน

iii. ความสามารถในการตรวจสอบลายเซ็นในแต่ละคน

iv. ความสามารถในการตรวจสอบตัวจริงของเอกสารว่ามีการแก้ไขหรือไม่

b. ความสามารถในด้านความปลอดภัย (Security)

i. การป้องกันการแก้ไขข้อมูลในแต่ละช่องของเอกสาร (Protect Field)

ii. การจำกัดสิทธิ์ในการลงนามและการดูเอกสารในแต่ละส่วนของเอกสาร

iii. ความสามารถในการติดตามการแก้ไขเอกสาร (Documents Tracking)

c. การติดตามเอกสารในแต่ละชุด (version) (ก่อนและหลังการลงนามในแต่ละคน)

d. ความสามารถในการแนบเอกสารประกอบ (Attachment)

i. การแนบเอกสารประกอบในรูปแบบต่างๆ (ข้อความ แฟ้มข้อมูล ภาพ)

ii. การตรวจสอบการเปลี่ยนแปลงและการแก้ไขของเอกสารที่แนบ

e. การ Authenticate จะใช้หลักการของ Public Key Encryption โดยใช้ RSA / MD5 เป็น Digital Signature

5) ออกแบบส่วนติดต่อกับผู้ใช้ (User Interface) ในการใช้งานเอกสารลงนามทางอิเล็กทรอนิกส์ เช่น การลงนามในเอกสาร การตรวจสอบลายเซ็นในเอกสาร เพื่อให้เอกสารที่ออกแบบง่ายต่อการใช้งาน จึงต้องมีการกำหนดส่วนในการติดต่อกับผู้ใช้ (user Interface) โดยมีฟังก์ชั่นหลักคือ

a. การลงนามทางอิเล็กทรอนิกส์ในแต่ละคน

b. การแก้ไขเอกสารในแต่ละส่วนของเอกสาร

c. การดูเอกสารในแต่ละชุด (version)

d. การเก็บข้อมูลของเอกสาร

e. การแนบเอกสารประกอบในแต่ละชนิด

6) ใช้ XML เป็นต้นแบบในการพัฒนาเอกสารทางอิเล็กทรอนิกส์เท่านั้น

7) การเชื่อมต่อกับ Browser ที่สามารถรับ XML เป็นหลัก

8) การวิจัยจะไม่ร่วมถึงการสร้างตัวเขียนภาษา (Editor) ของ XML ใหม่ที่ออกแบบ

9) การศึกษานี้มีการสร้างตัวอย่างของแบบฟอร์มอิเล็กทรอนิกส์เพื่อใช้ในการทดสอบรูปแบบของ XML ที่ออกแบบ (Prototype)

10) การทำงานร่วมกันเป็นแบบ Asynchronous Cooperative กล่าวคือ ไม่ได้ทำงานพร้อมกันหลายๆ คนในเวลาเดียวกัน แต่เป็นการทำงานร่วมกันทีละคน

11) รูปแบบเอกสารที่ออกแบบไม่รวมถึงขั้นตอนและหน้าที่การทำงานของระบบทางเดินเอกสาร (Document Workflow) เช่น การควบคุมเอกสารจากคนหนึ่งไปอีกคนหนึ่ง การตรวจเช็คการมาของเอกสาร

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. รูปแบบเอกสารที่ออกแบบสามารถนำมาใช้กับเอกสารที่ต้องการการลงนามร่วมกันหลายคนโดยผ่านทางคอมพิวเตอร์

2. สามารถนำมาใช้กับการทำงานร่วมกันของเอกสารบนอินเทอร์เน็ตได้ เช่น การแก้ไขเอกสารร่วมกันบนอินเทอร์เน็ต ซึ่งจะเหมาะกับกลุ่มการทำงานที่อยู่ห่างไกลกัน

3. เป็นพื้นฐานสำหรับโปรแกรมอื่นที่ต้องการรูปแบบเอกสารการลงนามทางอินเทอร์เน็ต เช่น EDI e-Commerce

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในโครงร่างวิทยานิพนธ์ฉบับนี้ได้มีการค้นคว้างานวิจัยที่เกี่ยวข้องกับการลงนามเอกสารทางอิเล็กทรอนิกส์ งานวิจัยที่ปัจจุบันทำอยู่ และปัญหาของงานวิจัยที่ทำมา ในหัวข้อต่อไปได้เสนอแนวทางคิด ในการแก้ปัญหาโดยการรวมแต่ละส่วนเข้าด้วยกัน เทคโนโลยีและทฤษฎีที่เกี่ยวข้องกับวิทยานิพนธ์ฉบับนี้ อาจแบ่งได้เป็นส่วนใหญ่ๆ คือ

2.1 งานวิจัยและเทคโนโลยีเกี่ยวกับรูปแบบเอกสารทางอินเทอร์เน็ต
2.2 งานวิจัยทางด้านการออกแบบรูปแบบเอกสารการลงนามทางอิเล็กทรอนิกส์ (Signed Document)
2.3 งานวิจัยทางด้านการติดตามการแก้ไขของเอกสาร (Document Versioning)
2.4 ทฤษฎีเกี่ยวกับการเข้ารหัสข้อมูลโดยใช้กุญแจแบบทั่วไป (Public Key Encryption) และการประยุกต์ใช้ลายเซ็นทางอิเล็กทรอนิกส์
2.5 ทฤษฎีการหาตัวแทนของข้อมูล (Hashing Function)

2.1 งานวิจัยและเทคโนโลยีเกี่ยวกับรูปแบบเอกสารทางอินเทอร์เน็ต

ปัจจุบันรูปแบบเอกสารที่นิยมใช้กันแพร่หลายในปัจจุบันคือ เอชทีเอ็มแอล (เอชทีเอ็มแอล - Hyper Text Markup Language) เนื่องจากมีการสร้างและโปรแกรมเบราว์เซอร์ (Browser) มากมายในตลาดรองรับ เช่น อินเทอร์เน็ตเอกซ์พลอเลอ (Internet Explorer) หรือ IE ของไมโครซอฟท์ นาวิเกเตอร์ (Navigator) ของบริษัท เนทสเคป แต่ตัว เอชทีเอ็มแอล ออกแบบมาเพื่อกำหนดรูปแบบการแสดงผลทางอินเทอร์เน็ตเท่านั้น ไม่สามารถบอกคุณลักษณะของข้อมูลที่อยู่ในแฟ้มข้อมูล ทำให้เอกสารทางอินเทอร์เน็ตไม่สามารถ แลกเปลี่ยนข้อมูลกันได้ระหว่างเอกสารมากกว่า 1 ชุด และผู้ใช้ไม่สามารถรู้คุณลักษณะของข้อมูลใน แฟ้มข้อมูลได้

จากข้อจำกัดของ เอชทีเอ็มแอล จึงได้มีการพัฒนารูปแบบของเอกสารทางอินเทอร์เน็ตโดยทางดับบลิวสามซี (W3C หรือ World Wide Web Consortium) เป็นแกนนำในการพัฒนารูปแบบเอกสารเอกซ์เอ็มแอล (XML หรือ Extensible Markup Language) รูปแบบเอกสารเอกซ์เอ็มแอลได้รับการพัฒนามาจากรูปแบบเอกสารเอสจีเอ็มแอล (SGML หรือ Standard Generalized Markup Language) ซึ่งได้รับการพัฒนาในปี 1986 เอสจีเอ็มแอล เป็นมาตรฐานในการกำหนดรูปแบบโครงสร้างของเอกสาร และเป็นมาตรฐาน ไอ เอส โอ 8879 (ISO8879) เอสจีเอ็มแอลได้รับการออกแบบมาเพื่อใช้กับเอกสารได้ทั่วๆ ไปไม่เจาะจงชนิด ของซอฟท์แวร์ รูปแบบของ เอสจีเอ็มแอลเป็นลักษณะที่เรียกว่าภาษามาร์คอัพ (Markup Language) กล่าวคือมีการใช้ตัวมาร์คอัพ (Markup) เพื่ออธิบายคุณลักษณะของข้อมูลระหว่างมาร์คอัพนั้น แต่เอสจีเอ็มแอลไม่ได้รับความแพร่หลาย

บนอินเทอร์เน็ต เนื่องจากโครงสร้างที่ซับซ้อนทำให้ใช้เวลามากในการประมวลผลและไม่มีเบราว์เซอร์ใดรองรับรูปแบบเอกสารเอสจีเอ็มแอลนี้

เอกซ์เอ็มแอลจึงได้ถูกพัฒนาเพื่อใช้งานบนอินเทอร์เน็ตโดยที่เอกซ์เอ็มแอลคือ สับเซ็ต (subset) ย่อยของเอสจีเอ็มแอล นั่นเอง โดยตัดสิ่งที่ไม่จำเป็นในการใช้งานบนอินเทอร์เน็ตออก และปรับปรุงให้มีรูปแบบที่ง่ายเพื่อให้ Browser สามารถรองรับได้โดยไม่ใช้ทรัพยากรของเครื่องมากเกินไป ปัจจุบันมีเบราว์เซอร์ ที่สนับสนุนการใช้งานเอกซ์เอ็มแอล เช่น อินเทอร์เน็ตเอกซ์พลอเลอร์ ชุด (version) 4 ของไมโครซอฟท์

เอกซ์เอ็มแอลเป็นภาษามาร์คอัพเช่นเดียวกับเอสจีเอ็มแอลซึ่งทำให้สามารถกำหนดรูปแบบโครงสร้างของ เอกสารได้ และบอกถึงความหมายของแต่ละส่วนในเอกสารได้เช่น บรรทัดไหนของแฟ้มข้อมูลเป็นข้อมูลเกี่ยวกับอะไร

จากการที่สามารถกำหนดรูปแบบของเอกสารโดยใช้เอกซ์เอ็มแอลทำให้สามารถกำหนด ส่วนของเอกสารได้ว่าช่วงใดของเอกสารที่มีการแก้ไข แก้ไขไปเป็นอะไรและสามารถใส่ข้อมูล เกี่ยวกับการลงนามทางอิเล็กทรอนิกส์ลงไปในตัวเอกสาร ในปัจจุบันเอกซ์เอ็มแอลเริ่มได้รับความนิยม อย่างแพร่หลายและมีหลายบริษัทที่ประกาศสนับสนุนเอกซ์เอ็มแอลในโปรแกรมของตน จึงเป็นไปได้ว่าเอกซ์เอ็มแอลจะมีบทบาทในอนาคตเช่นเดียวกับเอชทีเอ็มแอลในปัจจุบัน

2.2 งานวิจัยด้านการออกแบบเอกสารการลงนามทางอิเล็กทรอนิกส์ (Signed Document)

ในปัจจุบันได้มีการเสนอภาษามาร์คอัพที่เกี่ยวข้องกับการลงนามทางอิเล็กทรอนิกส์ (Signed Document Markup Language – SDML) ให้แก่ดับบลิวสามซีจากไฟแนนเชียล เซอร์วิส เทคโนโลยี คอนซอร์เตียม (Financial Services Technology Consortium) โดยเอกสารการเสนอเอสดีเอ็มแอล (SDML) อยู่ในการพิจารณาของคณะกรรมการดับบลิวสามซีอยู่ ในงานวิจัยของเอสดีเอ็มแอล นี้มีจุดประสงค์เพื่อพัฒนารูปแบบเอกสารทางอิเล็กทรอนิกส์เพื่อใช้ในงานที่ต้องมีการเซ็นรับรอง เช่น การสั่งจ่ายเช็คเอสดีเอ็มแอลได้ออกแบบโดยยึดเอสจีเอ็มแอลเป็นหลัก ทำให้ยังไม่สามารถนำมาใช้งานบนอินเทอร์เน็ตได้ ดังที่กล่าวไปแล้วว่าเอสจีเอ็มแอลเป็นรูปแบบที่ค่อนข้างซับซ้อนและยังไม่มีโปรแกรมเบราว์เซอร์ที่สนับสนุนเอสจีเอ็มแอลและเอสดีเอ็มแอล ไม่ได้วิจัยถึงในส่วนของการลงนามร่วมกันของหลายๆ คน ซึ่งมีการแก้ไขเอกสารในแต่ละส่วนที่ ตนเองมีอำนาจ

ตัวอย่างของรูปแบบเอกสาร SDML (Kravitz, 1998)

```
<sdml-doc docname="doc87" type="sample">
<action>
        <blkname>act1
        <crit>true
        <vers>1.0
        <function>sample
        <reason>process
</action>
<attachment>
        <blkname>att0123
        <adata encoding="text">
        This is a sample attachment
        </adata>
        </attachment>
<signature>
        <blkname>sig7
        <crit>true
        <vers>1.0
        <sigdata>
                <blockref>act1
                <hash alg="sha">278B7F348EECE3822A48C4D197FD5B920001C2E8
                <blockref>att0123
                <hash alg="sha">BC59D2FE5566F506910C5020B628E4136E1C6B39
                <nonce>9D9BC5AA75
                <sigref>cert-111111111-00000001
                <algorithm>sha/dsa
                <location>us
        </sigdata>
        <sig>
        2489E1E376F5CD823274010B0A6028
        EA3F2763F2:290B95F8F02CF6616B9
        C3A03DF0B50295A162295
</signature>
<cert>
```

<blkname>cert-111111111-00000001
<crit>true
<vers>1.0
<certtype>x509v1
<certissuer>/C=US/ST=NY/O=FSTC/OU=NYCA/
<certserial>1
<certdata>
308201F0308201B
.....
..... 910CE325DB7E
</cert>
</sdml-doc>

จากตัวอย่างข้างต้นจะเห็นว่าเอสดีเอ็มแอลมีแทก (TAG) ที่ออกแบบเพื่อ ใช้ในเอกสารที่มีการลงนาม เช่น

<action> </action> ใช้ในการกำหนดวิธีที่จะปฏิบัติกับเอกสาร (Function)

<signature></signature> ใช้ในการกำหนดส่วนข้อมูลที่ใช้ในการลงนามอิเล็กทรอนิกส์ เช่น Algorithm ข้อมูลลายเซ็น

<cert></cert> ใช้ในการกำหนดข้อมูลตรวจสอบของลายเซ็น (Signature Certification)

<attachment></attachment> กำหนดส่วนและชนิดของเอกสารที่แนบ

<message></message> ใช้ในการบอกถึงผลของการประมวลผลเอกสาร

2.3 งานวิจัยทางด้านการติดตามการแก้ไขของเอกสาร (Document Versioning)

งานวิจัยเกี่ยวกับการติดตามการแก้ไขของเอกสารเป็นการวิจัยที่เกี่ยวกับการออกแบบภาษามาร์คอัพที่ใช้ในการติดตาม ชุด (version)ของเอกสารเรียกว่าวีทีเอ็มแอล (VTML หรือ Versioned Text Markup Language) [18] ในงานวิจัยนี้จะวิจัยในเรื่องการหาชุด (version)ของเอกสารในแต่ละชุด (version)ที่ได้รับการแก้ไข ทำให้สามารถติดตาม ได้ว่าเอกสารก่อนและหลังการแก้ไขเป็นอย่างไร ซึ่งทำให้สามารถติดตามดูเอกสารที่มีการแก้ไขร่วมกัน หลายๆ คนได้ว่าใครเป็นคนแก้ไขที่ไหนและเมื่อไร

2.4 ทฤษฎีเกี่ยวกับการเข้ารหัสข้อมูลโดยใช้กุญแจแบบทั่วไป (Public Key Encryption)

การเข้ารหัสข้อมูลโดยใช้กุญแจแบบทั่วไปถูกเสนอครั้งแรกในปี 1976 โดย Diffie และ Hellman ข้อดีของการเข้ารหัสข้อมูลโดยใช้กุญแจแบบทั่วไปคือ

- กุญแจแบบทั่วไป (Public Key) อิงอยู่กับการคำนวณทางคณิตศาสตร์มากกว่าการ กระทำกับบิตของข้อมูล ทำให้การถอดรหัสนั้นยากกว่าการใช้การกระทำกับบิต (Bit Operation)
- กุญแจแบบทั่วไปใช้ กุญแจสองตัวแยกกันในการเข้าและถอดรหัสของข้อมูล ทำให้ลดปัญหาในความปลอดภัยของกุญแจและตรวจสอบได้ถึงเจ้าของของข้อมูลที่ส่งมา

ข้อเสียของกุญแจแบบทั่วไปคือเวลาที่ใช้นานกว่าวิธีแบบดั้งเดิม (Convention) ซึ่งอาจใช้ไม่ได้ในทางปฏิบัติและ การทำงานบางอย่าง นอกจากนี้ยังมีความยุ่งยากในการบำรุงรักษา กุญแจ 2 ตัว

2.4.1 การเข้ารหัสแบบอาร์เอสเอ (RSA Encryption)

กุญแจแบบทั่วไปที่ได้รับความนิยมมากที่สุดคืออาร์เอสเอ (RSA) สร้างโดย Ronald Rivest, Adi Sharma, และ Len Adleman ในปี ค.ศ. 1977 อาร์เอสเอใช้หลักของการหาเลขจำนวนเฉพาะ 2 จำนวนเป็นกุญแจในการเข้ารหัส (encrypt) และ ถอดรหัส (decrypt) ข้อมูลอาร์เอสเอยังใช้ในการคูณ หารและการหาเศษของการหารในการคำนวณ ซึ่งต่างจากวิธีของดีอีเอส (DES) ในการใช้การกระทำกับบิต (Bit Pattern Operation) จุดยากในการแกะรหัสอาร์เอสเอคือ การหาจำนวนเฉพาะ 2 ตัว ซึ่งอาจมีถึง 200 บิท ในแต่ละตัว ซึ่งเมื่อมาคูณกันจะเป็นเลขจำนวนเฉพาะ 400 บิท ซึ่งค่อนข้างยากมากในการสุ่มหาจำนวนเฉพาะ 2 จำนวนมาใช้ในการเข้าและถอดรหัส

2.4.2 กระบวนการทำงานแบบอาร์เอสเอ (RSA Algorithm)

ส่วนประกอบสำคัญของกระบวนการทำงานแบบอาร์เอสเอประกอบด้วยส่วนกุญแจเบบทั่วไปและกุญแจเฉพาะบุคคล (Private Key) ส่วนของกุญแจแบบทั่วไปประกอบด้วยตัวเลข 2 จำนวน คือ n, e โดยที่

$$n = pq$$

p และ q คือเลขจำนวนเฉพาะใดๆ โดย p และ q มีความยาวของหลักเท่ากัน e หาได้จากการสุ่มเลือกจำนวนใดๆ ที่มีตัวหารร่วมมากระหว่าง e กับ (p-1)(q-1) = 1 ในส่วนของ Private Key (d) หาได้จาก

$$d = e^{-1} \bmod((p-1)(q-1))$$

2.4.3 การประยุกต์ใช้งานอาร์เอสเอ

อาร์เอสเอสามารถนำไปประยุกต์ใช้ในการป้องกันข้อมูลถูกขโมยหรือการเปลี่ยนแปลงข้อมูลได้ ในหัวข้อนี้ขอยกตัวอย่างการนำเอากระบวนการทำงานแบบอาร์เอสเอไปใช้งานจริงในทางปฏิบัติ

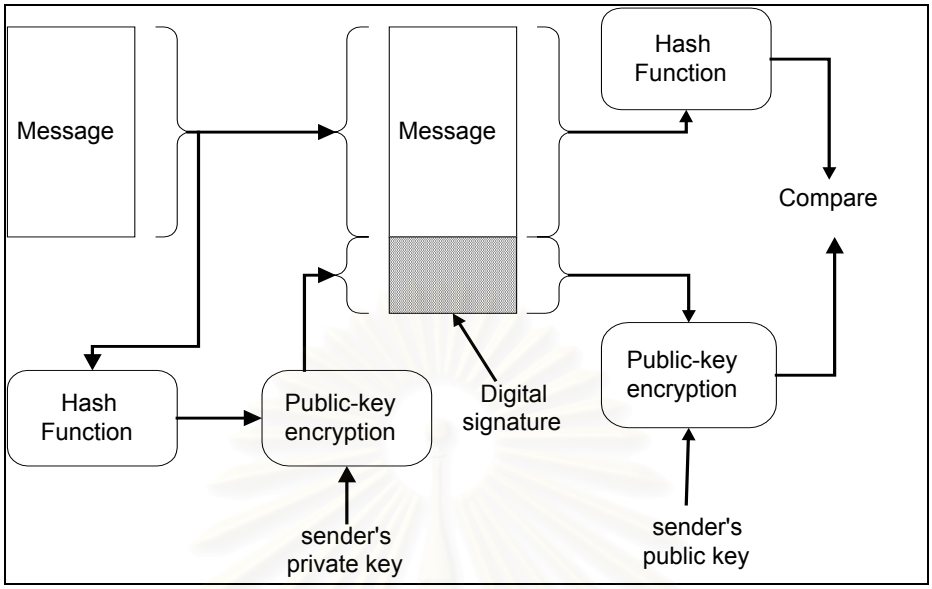
2.4.3.1 ลายเซ็นแบบดิจิตอล (Digital Signature)

ลายเซ็นแบบดิจิตอลคือโปรแกรมที่ทำการตรวจสอบว่าเอกสารที่ส่งมาถูกส่งโดยใคร และไม่มีการเปลี่ยนแปลงของเอกสารตั้งแต่ที่มีการเซ็นรับรองเอกสาร

การพัฒนาลายเซ็นแบบดิจิตอลมีอีกขั้นตอนวิธี(Algorithm) หนึ่งเข้ามาเกี่ยวข้อง คือ การย่อยข้อมูล (Message Digest) ซึ่งโดยสรุปแล้วคือการสร้างข้อมูลชุดหนึ่ง (Hash Code) จากข้อมูลทั้งหมดเพื่อแสดงถึงลักษณะของข้อมูลทั้งหมด ค่าแฮช (Hash Code) มีลักษณะที่สั้นกว่าข้อมูลทั้งหมด โดยปกติแล้วขั้นตอนการทำลายเซ็นแบบดิจิตอลกระทำได้โดย

- การหาค่าแฮชจากข้อความต้นฉบับและเข้ารหัสข้อมูลด้วยกุญแจส่วนบุคคลของผู้ส่ง หลังจากนั้นผู้ส่งทำการส่งข้อความต้นฉบับพร้อมด้วยค่าแฮชที่เข้ารหัสแล้วไปยังผู้รับ ทั้ง 2 ส่วนนี้เรียกว่าข้อความที่ถูกลงลายเซ็นดิจิตอล (Digitally Signed Message)
- ผู้รับถอดรหัสค่าแฮชด้วยกุญแจทั่วไป (Public Key) ของผู้ส่ง ผู้รับทำการสร้างค่าแฮชจากข้อความต้นฉบับ ผู้รับทำการเปรียบเทียบค่าแฮชที่สร้าง จากข้อความต้นฉบับกับที่ถอดรหัสจากผู้ส่ง ถ้าข้อความ 2 ชุดตรงกัน แสดงว่าข้อมูลถูกส่งโดยผู้ส่งจริง และข้อมูลไม่มีการแก้ไข

โปรดสังเกตว่าข้อมูลที่ส่งมาจะสามารถตรวจสอบโดยใครก็ได้ ผู้ตรวจสอบเพียงแต่ต้องรู้กุญแจทั่วไป (Public Key) ของผู้ส่งเท่านั้น



รูปที่ 2.1: กระบวนการสร้างและการตรวจสอบลายเซ็นแบบดิจิตอล

2.5 **ทฤษฎีการหาตัวแทนข้อมูล** (Hash Function)

ในการเข้ารหัสแบบอาร์เอสเอจะเห็นว่าถ้าข้อมูลมีจำนวนมาก การใช้วิธีอาร์เอสเอทำได้ช้าจึงมีผู้คิดที่จะเข้ารหัสข้อมูลเพียงบางส่วนของข้อมูลเท่านั้นเพื่อทำให้สามารถทำงาน
ได้เร็วขึ้น ได้มีผู้คิดทฤษฎีในการหาตัวแทนของข้อมูล (Hash Function) เพื่อใช้ในการลดเวลาเข้ารหัสข้อมูล อีกทั้งยังมีประโยชน์ในการทำลายเซ็นอิเล็กทรอนิกส์ ดังที่กล่าวไว้ในหัวข้อ 4.4.3.1 ที่ผ่านมา

โดยทั่วไปฟังก์ชั่นแฮชต้องมีคุณสมบัติที่สำคัญ 5 ประการดังต่อไปนี้ [17]

1. ต้องสามารถทำงานได้บนข้อมูลขนาดใดๆ
2. ต้องให้ค่าตัวแทนข้อมูล (Hash Code) ที่มีขนาดคงที่
3. ง่ายต่อการคำนวณหาค่าแฮช เพื่อใช้งานได้จริงในทางปฏิบัติ
4. ค่าแฮชต้องมาจากค่าข้อมูลทั้งหมด เพื่อเป็นตัวแทนของข้อมูลทั้งหมดได้
5. ยากต่อการคำนวณกลับหาค่าข้อมูลจริงจากค่าแฮชและต้องยากต่อการสร้างค่าแฮชที่เหมือนกันจากข้อมูลที่ต่างกัน 2 ชุด

ถ้าค่าแฮชที่หามาได้มีค่า 8 บิต จะสามารถแสดงข้อมูลที่แตกต่างได้เพียง 256 ชุด ซึ่งเห็นได้ว่าไม่พอเพียงในการเป็นตัวแทนข้อมูลได้ แต่ถ้าเป็น 32 บิต จะเห็นว่าสามารถมีได้ถึง 4 พันล้านชุดของข้อมูล โดยปกติแล้วค่าแฮชขนาด 128 บิต ถือได้ว่ามีความปลอดภัยสูง [17]

จะเห็นว่าการหาตัวแทนของข้อมูล (Hash Function) ไม่ได้เป็นความลับ ทุกคนสามารถรู้ได้ แต่การที่จะสร้างข้อมูลปลอมชุดหนึ่งเพื่อให้ได้ค่าตัวแทนข้อมูลที่เหมือนกับชุดเดิมเป็นการกระทำที่ยาก ดังนั้นการหาตัวแทนของข้อมูล (Hash Function) จึงมีประโยชน์ ในการตรวจสอบว่าข้อมูลที่ได้รับมีการเปลี่ยนแปลง หรือแก้ไขหรือไม่

2.6 แนวทางวิจัยและพัฒนา

จากทฤษฎีและงานวิจัยที่เกี่ยวข้องในหัวข้อที่ 2.1 – 2.5 ที่ใช้ในวิทยานิพนธ์ฉบับนี้สามารถนำมาวิจัยและพัฒนาต่อเพื่อแก้ปัญหาที่กล่าวไว้ในหัวข้อที่ 1.1 ได้ กล่าวคือ

ตารางที่ 2.1: แสดงความสัมพันธ์ระหว่างปัญหาและแนวทางวิจัย

ปัญหา	แนวทางวิจัยและพัฒนา
1. ทำอย่างไรเมื่อมีคนลงนามร่วมกันมากกว่า 1 คนในกรณีที่เป็นการเซ็นร่วมกัน และเซ็นตามลำดับสายงาน เช่น หัวหน้าแผนกไปยังผู้จัดการ	ใช้ลายเซ็นแบบดิจิตอล (Digital Signature) ในการลงนามทางอิเล็กทรอนิกส์ และออกแบบรูปแบบเอกสารทางอิเล็กทรอนิกส์ ในการกำหนดส่วนของเอกสารที่มีการป้องกันว่า ใครมีสิทธิ์แก้ไขส่วนใด รวมถึงส่วนของเอกสารที่มีการ บอกถึงลำดับของอำนาจในการลงนามโดยใช้รูปแบบของ เอกซ์เอ็มแอล (XML) เป็นต้นแบบในการพัฒนา
2. ทราบได้อย่างไรว่าคนที่เซ็นชื่ออนุมัตินั้นเป็นบุคคลนั้นจริง ไม่ใช่ผู้แอบอ้าง เนื่องจากเป็นการกระทำทางอิเล็กทรอนิกส์ ซึ่งข้อมูลสามารถปลอมแปลงได้ง่ายกว่าการเซ็นชื่อลงบนกระดาษ	ใช้ทฤษฎีลายเซ็นแบบดิจิตอล (Digital Signature) และการเข้ารหัสข้อมูลโดยใช้กุญแจทั่วไป (Public Key Encryption) ในการหาตัวจริงของผู้เซ็นเอกสาร
3. ไม่มีการเปลี่ยนแปลงเอกสารหลังจากมีการอนุมัติแล้ว ทำอย่างไรจึงแน่ใจได้ว่าเอกสาร ไม่มีการแก้ไขต่อเติม เมื่อมีการเซ็นชื่อทาง อิเล็กทรอนิกส์แล้ว	ใช้ทฤษฎีการหาตัวแทนของข้อมูล (Hash function) ในการตรวจสอบว่า เอกสารไม่มีการแก้ไข เพราะถ้ามีการแก้ไขค่าแฮช (Hash code) จะมีค่าเปลี่ยนไป

ตารางที่ 2.1: แสดงความสัมพันธ์ระหว่างปัญหาและแนวทางวิจัย (ต่อ)

ปัญหา	แนวทางวิจัยและพัฒนา
4. ในกรณีที่มีผู้อนุมัติหลายคน เอกสารต้องมีการผ่านมากกว่าหนึ่งคนขึ้นไป และคนที่อนุมัติ ถัดมามักมีความสามารถในการแก้ไขเอกสารได้ ปัญหาที่เกิดคือตรวจสอบได้อย่างไรว่าผู้เซ็นชื่อก่อนรับผิดชอบเฉพาะส่วนที่ตัวเองแก้ไขเพิ่มเติม ไม่ต้องรับผิดชอบในส่วนที่ผู้อนุมัติคนอื่นเข้าไปแก้ไขเพิ่มเติม	การใช้ชุด (version)ของเอกสาร (Document Versioning) เพื่อติดตามว่า เอกสารก่อนที่ถูกแก้ไขเป็นอย่างไร และถูกแก้ไข โดยใครโดยใช้ลายเซ็นแบบดิจิตอล (Digital Signature) เป็นตัวบ่งชี้ ผู้ที่ทำการแก้ไข
5. การแนบเอกสารประกอบเมื่อมีการเซ็น และเอกสารประกอบต้องไม่สามารถแก้ไขหรือเปลี่ยนแปลง เมื่อมีการเซ็นเกิดขึ้น	ใช้การหาตัวแทนของข้อมูล (Hash Function) กับเอกสารประกอบเพื่อป้องกัน การแก้ไขข้อมูลหลังจากมีการลงนามในเอกสาร
6. ต้องการดูเอกสารก่อนที่มีการเซ็นและแก้ไขโดยแต่ละคน หรือส่วนที่มีการแก้ไขของแต่ละคน	ใช้ชุด (version)ของเอกสาร (Document Versioning) ในการติดตามการ แก้ไขของเอกสารของแต่ละคน

2.7 ขั้นตอนการสร้างระบบ

ในการสร้างระบบเพื่อใช้ลายเซ็นดิจิตอลในการลงนามร่วมกันสามารถแบ่งขั้นตอนออกเป็น 4 ส่วนที่สำคัญดังต่อไปนี้

1. การวิเคราะห์ความต้องการ (Requirement Analysis)
2. การออกแบบโครงสร้างเอกสารที่มีการลงนาม
   a. คุณสมบัติทางการทำงาน (Functional Specification)
   b. พฤติกรรมและการทำงานร่วมกันของส่วนต่างๆในเอกสาร
3. การพัฒนาโดยสร้างเอกสารจากข้อ II โดยใช้รูปแบบเอกซ์เอ็มแอล (XML Format)
4. การพัฒนาระบบเพื่อรองรับการใช้งานของเอกสารในข้อ III และการเชื่อมต่อในอนาคต

ในแต่ละขั้นตอนมีรายละเอียดที่สำคัญดังต่อไปนี้

1. การวิเคราะห์ความต้องการ (Requirement Analysis) คือการวิเคราะห์ความต้องการของผู้ใช้ ในการลงนามในกระดาษแบบเดิม โดยศึกษาว่าขั้นตอนและวิธีการลงนามโดยใช้กระดาษมีความต้องการ ด้านใดบ้าง และเมื่อนำมาใช้กับลายเซ็นดิจิตอลในรูปของแบบฟอร์มทางอิเล็กทรอนิกส์มีอะไรบ้าง ที่ต้องพิจารณา เช่น การตรวจสอบลายเซ็น การลงนามตามลำดับขั้น

2. การออกแบบโครงสร้างเอกสารโดยใช้ความต้องการในข้อ 1) เป็นหลักในการออกแบบ คือการออกแบบโครงสร้างของเอกสารเพื่อรองรับความต้องการที่ได้จากข้อ 1) โดยโครงสร้างที่ ออกแบบเป็นเพียงข้อ

กำหนด (Specification) ที่ควรมีเพื่อรองรับความต้องการทางด้าน การลงนามในทางอิเล็กทรอนิกส์โดยไม่ขึ้นกับรูปแบบของเอกสารใดเป็นหลักในการออกแบบ

3. การพัฒนาเอกสารจาก 2) โดยใช้เอกซ์เอ็มแอล (XML) ในขั้นตอนนี้เป็นการสร้างเอกสารเอกซ์เอ็มแอล (XML) โดยใช้โครงสร้างที่ออกแบบในข้อ 2) เป็นตัวกำหนดถึงแทกและเขตข้อมูล (field)ต่างๆ ในเอกซ์เอ็มแอล (XML) ในการพัฒนาเลือกใช้เอกซ์เอ็มแอล (XML) เป็นหลัก เนื่องจาก เอกซ์เอ็มแอล (XML) เป็นเอกสารที่สามารถกำหนดคุณลักษณะของข้อความ (contents) ในเอกสารได้ ซึ่งสะดวกในการนำมาสร้างเอกสารที่เป็นการลงนามทางดิจิตอลได้ดีกว่าเอกสารในรูปแบบอื่น เช่น ไมโครซอฟท์เวิร์ด (MS WORD) ที่รูปแบบของเอกสารเน้นในด้านการแสดงรูปแบบต่างๆ ทางจอ (Presentation) ในขั้นตอนนี้ทำให้สามารถตรวจสอบได้ว่าโครงสร้างที่ออกแบบใน 2) นั้นมีปัญหาอะไรบ้างในการพัฒนา บนเอกสารใน รูปแบบต่างๆ แต่ในวิทยานิพนธ์ฉบับนี้เน้นที่เอกซ์เอ็มแอล (XML) เพียงอย่างเดียว

4. การพัฒนาระบบเพื่อรองรับการใช้งานของเอกสารในข้อ 3) หลังจากที่ได้สร้างเอกสารเอกซ์เอ็มแอล (XML) ตามโครงสร้างที่ออกแบบมาแล้วในข้อ 2) ขั้นต่อไปคือการสร้างระบบเพื่อใช้งานเอกสารที่สร้างในส่วนนี้ต้องมีการออกแบบคุณสมบัติโครงสร้างของระบบที่รองรับการใช้งานเอกสารที่มีการลงนาม (System Architecture) และการสร้างส่วนติดต่อกับผู้ใช้ (User Interface) เพื่อให้ผู้ใช้สามารถใช้งานเอกสารที่ สามารถลงนามได้สะดวก และเป็นการสร้างโปรแกรมประยุกต์เพื่อทดสอบโครงสร้างเอกสารที่ออกแบบ

บทที่ 3

ความต้องการของระบบงาน

3.1 ความต้องการของระบบงานเอกสารที่ต้องมีการลงนามในแง่ของผู้ใช้ (User Requirement)

จากการศึกษาการใช้แบบฟอร์มของหน่วยงานราชการ (สถาบันวิทยบริการ จุฬาลงกรณ์วิทยาลัย) ในส่วนของเอกสารที่มี การลงนาม จะพบว่าเอกสารส่วนใหญ่เป็นแบบลำดับชั้น และมีความต้องการการลงนามที่ต่างกัน ทั้งนี้จากการสัมภาษณ์เจ้าหน้าที่ที่ใช้งานแบบฟอร์มและการสังเกต จะพบว่าอาจสามารถแบ่ง เป็นประเภทใหญ่ๆ ได้ดังนี้คือ

1. เอกสารที่มีการลงนามคนเดียว ตัวอย่างเช่น เช็คที่มีผู้มีอำนาจสั่งจ่ายคนเดียว สลิปบัตรเครดิต
2. เอกสารที่มีการลงนามตั้งแต่ 2 คนขึ้นไป สามารถแบ่งได้เป็น 3 ลักษณะคือ

2.1. เอกสารที่มีการลงนามแบบลำดับชั้น ต้องมีการกำหนดว่าใครเป็นผู้ลงนามก่อนหลัง เช่น หนังสืออนุมัติเบิกจ่าย ใบเบิกพัสดุ เอกสารราชการส่วนใหญ่เป็นแบบนี้

2.2. เอกสารที่มีการลงนามแบบไม่มีลำดับชั้น (Peer to Peer) คือใครเป็นผู้ลงนามก่อนหลังก็ได้ เช่น เอกสารรับรองรายงานการประชุม เช็คที่มีผู้มีอำนาจสั่งจ่าย 2 คน หรือเอกสารที่มีกรรมการมากกว่า 1 คน เอกสารแบบนี้ลำดับการลงนามจะไม่สำคัญ

2.3. เอกสารที่มีการผสมทั้งแบบ 2.1 และ 2.2 เช่นรายงานการตรวจการจ้าง กรรมการคนใดเซ็นก่อนหรือหลังก็ได้ ประธานกรรมการเป็นคนอนุมัติขั้นสุดท้าย

จากประเภทของเอกสารที่มีการลงนามขั้นต้น ทำให้สามารถหาความต้องการหลักในการ ใช้ลายเซ็นดิจิตอลเพื่อลงนามในเอกสารได้ดังต่อไปนี้

1. ความต้องการของเอกสารที่มีการลงนามคนเดียว

1.1. ไม่มีการเปลี่ยนแปลงเอกสารเมื่อมีการลงนามเกิดขึ้น

1.2. สามารถตรวจสอบตัวจริงของลายเซ็นได้ว่าใครเป็นผู้เซ็น

1.3. เมื่อมีการแก้ไขเอกสารสามารถลงลายมือชื่อกำกับเช่นเดียวกับการแก้ไขเอกสารทางกระดาษ

1.4. มีการแนบเอกสารรับรองได้

1.5. มีการ ทำตราประทับเอกสาร

1.6. การลงลายเซ็นที่ไม่มีช่องให้ใส่ เช่น รับรองสำเนา

1.7. การรับรองเฉพาะแถว เช่น บัญชีคุมวัสดุ

1.8. การแยกระหว่างเอกสารต้นฉบับ (Read/Write) และเอกสารสำเนา (Read Only)

2. ความต้องการของเอกสารที่มีการลงนามตั้งแต่ 2 คนขึ้นไป จะรวมความต้องการของเอกสารที่มีการลงนามคนเดียวกับลักษณะเฉพาะของการลงนามตั้งแต่ 2 คน โดยสามารถแบ่งได้ 2 ลักษณะ คือ

2.1. เอกสารที่มีการลงนามแบบลำดับชั้น มีความต้องการดังนี้คือ

2.1.1. สามารถกำหนดส่วนรับผิดชอบในแต่ละเขตข้อมูล (Protected Field) ในแต่ละคน ว่าใครมีอำนาจแก้ไขในส่วนใด

2.1.2. กำหนดส่วนของการแนบได้ว่าใครแนบเอกสารอะไร

2.1.3. มีการกำหนดต้นไม้แสดงสิทธิ์ (Tree of Authorize) ในการกำหนดลำดับของการลงนาม

2.1.4. ลายเซ็นคนเดียวอาจมีลำดับการลงนามไม่เหมือนกัน เช่น ผู้ขอเบิกเงินและผู้รับเงิน

2.1.5. ต้องสามารถเปลี่ยนแปลงอำนาจและลำดับการลงนามได้ (Authorization Flow)

2.2. เอกสารที่มีการลงนามแบบไม่มีลำดับชั้น (Peer to Peer) ความต้องการของเอกสารชนิดนี้คล้ายกับ 2.1 เพียงแต่ต้นไม้แสดงสิทธิ์ (Tree of Authorize) อยู่ในระดับเดียวกัน ใครลงนามก่อนหลังก็ได้

ที่กล่าวมาคือความต้องการโดยกว้างๆ ในการใช้ลายเซ็นดิจิตอลกับเอกสารอิเล็กทรอนิกส์แต่ในทางปฏิบัติจริงยังมีความต้องการอื่นที่เกี่ยวข้องกับการสร้าง แก้ไข ตรวจสอบ การเก็บรักษาชุด (version) ของเอกสารเพื่อเสริมให้การทำงานสมบูรณ์ โดยสามารถแบ่งความต้องการอื่นๆ ได้ดังนี้

1. ความต้องการทางด้านลายเซ็นดิจิตอล

1.1. การตรวจสอบลายเซ็นในเอกสารกับตัวจริงของบุคคลที่เซ็น (Authentication)

1.2. ความสามารถในการเลือกขั้นตอนวิธี (Algorithm) ในการทำลายเซ็นแบบดิจิตอล (Digital Signature)

2. ความต้องการทางด้านส่วนติดต่อกับผู้ใช้ (User Interface)

2.1. รูปภาพของลายเซ็นแบบดิจิตอล (Digital Signature) เพื่อแสดงการลงลายเซ็น

2.2. ความต้องการฟังก์ชั่นในการสร้าง ลบ แก้ไข ตรวจสอบ ลงลายเซ็น การแนบเอกสารสำหรับผู้ใช้งาน

2.3. ความต้องการภาษาที่ง่ายในการอธิบายเอกสารและสามารถขยายได้เพื่อรองรับ ความต้องการใหม่

3. ความต้องการด้านการตรวจสอบชุด (version)ของเอกสารเพื่อตรวจสอบว่าเอกสารก่อนที่มีการเซ็นชื่อมีการแก้ไขอะไรบ้าง

3.1. การเก็บบันทึกเอกสารในแต่ละชุด (version)ก่อนที่มีการเปลี่ยนแปลง

3.2. การเรียกดูเอกสารในแต่ละชุด (version)ก่อนที่มีการแก้ไขหรือลงลายเซ็นตามลำดับ

4. การไหลของเอกสาร (Document Flow) คือการตรวจสอบว่าเอกสารควรส่งให้ใครต่อ หลังจากมีการเซ็นเกิดขึ้น

4.1. การระบุชื่อของผู้ส่งคนต่อไปเพื่อกระทำกับเอกสาร

4.2. การตรวจสอบว่าเอกสารมีการเซ็นไปแล้วกี่คน ขาดอีกกี่คน

5. ความต้องการด้านความปลอดภัย (Security)

5.1. มีการกำหนดกลุ่มของผู้มีสิทธิ์ในลักษณะของ ACL (Access Control List) สำหรับแต่ละเขตข้อมูลในแบบฟอร์ม

5.2. กำหนดสิทธิ์ของผู้มีลายเซ็นและอำนาจในการแก้ไขเอกสาร (Security Policy)

5.3. กำหนดสิทธิ์ในการแสดงข้อมูลว่าใครมีสิทธิ์ดูได้บ้างโดยขึ้นอยู่กับแบบของเอกสาร เช่น ใบแจ้งเงินเดือน อาจแสดงเงินเดือนให้ดูได้เฉพาะเจ้าของและหัวหน้า แต่ไม่แสดงให้ดูสำหรับผู้ที่ไม่มีสิทธิ์

5.4. การกำหนดอำนาจตามบทบาท (Role) และอำนาจตามบุคคล

5.5. อำนาจการลงนามขึ้นกับเวลา และบุคคลที่ลงนามมีการเปลี่ยนแปลงไป เช่น กรรมการตรวจสอบเฉพาะกิจ มีอำนาจ 6 เดือน

5.6. การกำหนดการเซ็นแทนในช่วงเวลา เช่นผู้มีอำนาจไม่อยู่ และมอบอำนาจให้ผู้อื่นเซ็นแทน

5.7. กำหนดกฎของอำนาจหน้าที่ (Rule of Authorization) ลายเซ็น (Signature) เช่นผู้มีอำนาจตามวงเงิน

5.8. กำหนดชนิดและความสำคัญของลายเซ็นแต่ละแบบในแต่ละคน เช่น ลายเซ็นแบบย่อใช้ในการเซ็นรับรู้เอกสาร เอกสารที่สำคัญอาจมีการเซ็นชื่อเต็มโดยที่ Algorithm ในการลงลายเซ็นอาจแตกต่างกันเพื่อความรวดเร็วในการลงลายเซ็น เนื่องจากลายเซ็นแบบดิจิตอลใช้เวลาในการลงลายเซ็นค่อนข้างนาน การลงลายเซ็นที่ไม่สำคัญอาจใช้รหัสผ่านเพียงอย่างเดียวเป็นการตรวจสอบ

3.2 ความต้องการของระบบเอกสารที่มีการลงนามของผู้พัฒนาและผู้ดูแลระบบ (System Requirement)

จากความต้องการของระบบในแง่ของผู้ใช้ สามารถมาวิเคราะห์ความต้องการในแง่ของผู้พัฒนาระบบ เนื่องจากการกำหนดรูปแบบเอกสารเป็นเพียงการกำหนดในแง่ของคุณลักษณะ (Specification) เท่านั้น แต่คุณลักษณะของเอกสารต้องสามารถนำมาพัฒนาในรูปแบบโปรแกรมได้ง่าย และสะดวกต่อ การเพิ่มขยายในอนาคต จากการวิเคราะห์และค้นคว้าเพิ่มเติมจาก XML-Signature Requirements W3C Working Draft 1999-August-20 สามารถสรุปความต้องการหลักๆ ได้ดังต่อไปนี้

1. สามารถใช้กับเอกสารที่อยู่ในรูปเอกซ์เอ็มแอล (XML) โดยมีการเปลี่ยนแปลงเอกสารเอกซ์เอ็มแอลเดิมน้อยที่สุด ทำให้สามารถประยุกต์ใช้กับเอกสารเอกซ์เอ็มแอลเดิมที่มีอยู่แล้ว

2. ต้องสามารถขยายความสามารถในด้านอื่นๆ ได้ในอนาคต เช่น งานด้านกระแสงาน (Workflow) การจัดการเอกสาร (Document Management)

3. สามารถนำมาพัฒนาโปรแกรมประยุกต์ได้ง่าย

4. สามารถกำหนดรูปแบบของขั้นตอนวิธีที่ใช้ในการทำลายเซ็นแบบดิจิตอล (Digital Signature) หรือกล่าวอีกนัยหนึ่งคือ รูปแบบเอกสารต้องไม่ยึดติดกับขั้นตอนวิธีใดโดยเฉพาะ

5. รองรับรูปแบบการพัฒนาในระบบเว็บและอินเทอร์เน็ต และสามารถรองรับการทำงานแบบกระจายการทำงาน (De-centralized) ได้ กล่าวคือ เอกสารที่มีการลงนามสามารถทำงานร่วมกันได้แม้จะอยู่ต่างเครื่องกัน

6. สามารถนำไปพัฒนาการเชื่อมต่อกับเว็บเบราว์เซอร์ (Web Browser) ที่มีอยู่ในปัจจุบัน

7. ใช้ได้กับการส่งเอกสารในอินเตอร์เน็ท เช่น Mail HTTP และ FTP

8. ความสามารถของระบบในการสร้างหรือแก้ไข และลงนามชุดเอกสารที่มีการลงนาม

9. ระบบต้องมีความสามารถในการตรวจสอบ (Audit) ได้ คือมีใครมาแก้ไข ลงนามเมื่อใด และแก้ไขส่วนใดบ้าง

10. มีความสามารถในการทำสำเนาเอกสารลงนาม (Backup) เพื่อป้องกันการสูญหายของเอกสาร โดยมีการลงนามกำกับเอกสารสำรองนั้นด้วย

11. มีความสามารถในการป้องกันระบบ (Security) นั่นคือป้องกันการแก้ไขชุดเอกสารในลักษณะรวม เนื่องจากเอกสารที่ออกแบบมีโครงสร้างเป็นตัวอักษร (Text) ซึ่งสามารถอ่านเข้าใจได้ ดังนั้น ระบบต้องมีความสามารถในการป้องกันคนภายนอกที่ไม่ได้รับสิทธิ์มาแก้ไข หรือดูเอกสารได้

12. ความสามารถในการกำหนดสิทธิ์การแก้ไขในแต่ละส่วนตามหน้าที่ความรับผิดชอบ เช่น ผู้ดูแลระบบ (System Administrator) ผู้ป้อนข้อมูล (Data Entry) หัวหน้า (Supervisor)

13. ระบบสามารถรองรับความผิดพลาดอันเกิดจากการใช้งาน ทั้งในแง่ของผู้ใช้ระบบ และตัวเอกสารเอง (Exception conditions) เช่น เอกสารไม่มีการลงนาม ต้องทำอย่างไร เอกสารมีข้อมูลไม่สมบูรณ์ จะแสดงข้อความผิดพลาดที่ใคร และใครจะเป็นผู้แก้ไข

14. มีระบบในการตรวจสอบเอกสารที่ถูกยกเลิกได้

3.3 ตารางสรุปความต้องการ (Requirement Specification)

ตารางที่ 3.1: สรุปความต้องการของระบบ

1.	ความต้องการของเอกสารที่มีการลงนามคนเดียว	1.1.	ไม่มีการเปลี่ยนแปลงเอกสารเมื่อมีการลงนามเกิดขึ้น
		1.2.	สามารถตรวจสอบตัวจริงของลายเซ็นได้ว่าใครเป็นผู้เซ็น
		1.3.	เมื่อมีการแก้ไขเอกสารสามารถลงลายมือชื่อกำกับเช่นเดียวกับการแก้ไขเอกสารทางกระดาษ
		1.4.	มีการ ทำตราประทับเอกสาร
		1.5.	การลงลายเซ็นที่ไม่มีช่องให้ใส่ เช่น รับรองสำเนา
		1.6.	การรับรองเฉพาะแถว เช่น บัญชีคุมวัสดุ
		1.7	การแยกระหว่างเอกสารต้นฉบับ (Read/Write) และเอกสารสำเนา (Read Only)
2.	ความต้องการของเอกสารที่มีการลงนามตั้งแต่ 2 คนขึ้นไป	2.1.1.	สามารถกำหนดส่วนรับผิดชอบ (Protected Field) ในแต่ละคน ว่าใครมีอำนาจแก้ไขในส่วนใด
		2.1.2.	ลายเซ็นคนเดียวอาจมีลำดับการลงนามไม่เหมือนกัน เช่น ผู้ขอเบิกเงินและผู้รับเงิน
		2.1.3.	มีการกำหนดต้นไม้แห่งการลงนาม (Tree of Authorization) ในการกำหนดลำดับของการลงนาม
		2.1.4.	ต้องสามารถเปลี่ยนแปลงอำนาจและลำดับการลงนามได้ (Authorization Flow)
3.	ความต้องการทางด้านลายเซ็นดิจิตอล	3.1.	การตรวจสอบลายเซ็นในเอกสารกับตัวจริงของบุคคลที่เซ็น (Authentication)
		3.2.	ความสามารถในการเลือกขั้นตอนวิธีในการทำลายเซ็นแบบดิจิตอล
4.	ความต้องการทาง User Interface (UI)	4.1.	รูปภาพของลายเซ็นแบบดิจิตอลเพื่อแสดงการลงลายเซ็น
		4.2.	ความต้องการฟังก์ชั่นในการสร้าง ลบ แก้ไข ตรวจสอบ ลงลายเซ็น การแนบเอกสาร

ตารางที่ 3.1: สรุปความต้องการของระบบ (ต่อ)

		4.3.	ความต้องการภาษาที่ง่ายในการอธิบายเอกสารและสามารถขยายได้เพื่อรองรับความต้องการใหม่
5.	ความต้องการด้านการตรวจสอบชุด (version)ของเอกสาร	5.1.	การเก็บบันทึกเอกสารในแต่ละชุด (version)ก่อนที่มีการเปลี่ยนแปลง
		5.2.	การเรียกดูเอกสารในแต่ละชุด (version)ก่อนที่มีการแก้ไขหรือการลงลายเซ็นตามลำดับ
6.	การไหลของเอกสาร (Document Flow)	6.1.	การระบุชื่อของผู้ส่งคนต่อไปเพื่อกระทำกับเอกสาร
		6.2.	การตรวจสอบว่าเอกสารมีการเซ็นไปแล้วกี่คน ขาดอีกกี่คน
7.	ความต้องการด้านความปลอดภัย (Security)	7.1.	มีการกำหนดกลุ่มของผู้มีสิทธิ์ในลักษณะของ ACL (Access Control List) สำหรับแต่ละเขตข้อมูลในแบบฟอร์ม
		7.2.	กำหนดสิทธิ์ของผู้มีลายเซ็นและอำนาจในการแก้ไขเอกสาร (Security Policy)
		7.3.	กำหนดสิทธิ์ในการแสดงข้อมูลว่าใครมีสิทธิ์ดูได้บ้างโดยขึ้นอยู่กับแบบของเอกสาร เช่น ใบแจ้งเงินเดือน อาจแสดงเงินเดือนให้ดูได้เฉพาะเจ้าของและหัวหน้า แต่ไม่แสดงให้ดูสำหรับผู้ที่ไม่มีสิทธิ์
		7.4.	การกำหนดอำนาจตามบทบาท และอำนาจตามบุคคล
		7.5.	อำนาจการลงนามขึ้นกับเวลา และบุคคลที่ลงนามมีการเปลี่ยนแปลงไป เช่น กรรมการตรวจสอบเฉพาะกิจ มีอำนาจ 6 เดือน
		7.6.	การกำหนดการเซ็นแทนในช่วงเวลา เช่นผู้มีอำนาจไม่อยู่ และมอบอำนาจให้ผู้อื่นเซ็นแทน
		7.7.	กำหนดกฎของอำนาจหน้าที่ (Rule of Authorization) และการลงลายเซ็น (Signature) เช่นผู้มีอำนาจตามวงเงิน
		7.8.	กำหนดชนิดและความสำคัญของลายเซ็นแต่ละแบบในแต่ละคน เช่น ลายเซ็นแบบย่อ ใช้ในการเซ็นรับรู้เอกสาร
8.	ความต้องการด้านการแนบเอกสาร	8.1.	มีการแนบเอกสารรับรองได้
		8.2.	กำหนดส่วนของการแนบเอกสารได้ว่าใครแนบเอกสารอะไร
9.	ความต้องการในแง่ผู้พัฒนาและผู้ดูแลระบบ (System Requirement)	9.1.	สามารถใช้กับเอกสารที่อยู่ในรูปเอกซ์เอ็มแอลโดยมีการเปลี่ยนแปลงเอกสารเอกซ์เอ็มแอลเดิมน้อยที่สุด ทำให้สามารถประยุกต์ใช้กับเอกสารเอกซ์เอ็มแอลเดิมที่มีอยู่แล้ว
		9.2.	ต้องสามารถขยายความสามารถในด้านอื่นๆ ได้ในอนาคต เช่น งานด้านกระแสงาน (Workflow) และการจัดการด้านเอกสาร (Document Management)
		9.3.	สามารถนำมาพัฒนาโปรแกรมประยุกต์ได้ง่าย

ตารางที่ 3.1: สรุปความต้องการของระบบ (ต่อ)

	9.4.	สามารถกำหนดรูปแบบของขั้นตอนวิธีที่ใช้ในการทำลายเซ็นแบบดิจิตอล หรือกล่าวอีกนัยหนึ่งคือ รูปแบบเอกสารต้องไม่ยึดติดกับขั้นตอนวิธีใดโดยเฉพาะ
	9.5.	รองรับรูปแบบการพัฒนาในระบบเว็บและอินเทอร์เน็ต และสามารถรองรับการทำงานแบบกระจายการทำงาน (De-centralized) ได้ กล่าวคือ เอกสารที่มีการลงนามสามารถทำงานร่วมกันได้แม้อยู่ต่างเครื่องกัน
	9.6.	สามารถนำไปพัฒนาการเชื่อมต่อกับเว็บเบราว์เซอร์ที่มีอยู่ในปัจจุบัน
	9.7.	ใช้ได้กับการส่งเอกสารในอินเทอร์เน็ต เช่น Mail HTTP FTP
	9.8.	ความสามารถของระบบในการสร้าง แก้ไข และลงนามชุดเอกสารที่มีการลงนาม
	9.9.	ระบบต้องมีความสามารถในการตรวจสอบ (Audit) ได้ คือมีใครมาแก้ไข ลงนาม เมื่อใด และแก้ไขส่วนใดบ้าง
	9.10.	มีความสามารถในการทำเอกสารลงนามสำรอง (Backup) เพื่อป้องกันการสูญหายของเอกสาร
	9.11.	มีความสามารถในการป้องกันระบบ (Security) นั่นคือป้องกันการแก้ไขชุดเอกสารในลักษณะรวม เนื่องจากเอกสารที่ออกแบบมีโครงสร้างเป็นตัวอักษร (Text) ซึ่งสามารถอ่านเข้าใจได้ ดังนั้น ระบบต้องมีความสามารถในการป้องกัน คนภายนอกที่ไม่ได้รับสิทธิ์มาแก้ไข หรือดูเอกสารได้
	9.12.	ความสามารถในการกำหนดสิทธิ์การแก้ไขในแต่ละส่วนตามหน้าที่ความรับผิดชอบ เช่น ผู้ดูแลระบบ ผู้ป้อนข้อมูล หัวหน้างาน
	9.13.	ระบบสามารถรองรับความผิดพลาดอันเกิดจากการใช้งาน ทั้งในแง่ของผู้ใช้ระบบ และตัวเอกสารเอง (Exception conditions) เช่น เอกสารไม่มีการลงนาม ต้องทำอย่างไร เอกสารมีข้อมูลไม่สมบูรณ์ จะแสดงข้อความผิดพลาดที่ใคร และใครจะเป็นผู้แก้ไข
	9.14.	มีระบบในการตรวจสอบเอกสารที่ถูกยกเลิกได้

จากการวิเคราะห์ความต้องการของเอกสารที่มีการลงนามในแบบต่างๆ นำไปสู่การออกแบบโครงสร้างเอกสารให้รองรับความต้องการของเอกสารที่มีการลงนาม ซึ่งจะกล่าวถึงในบทต่อไป

บทที่ 4

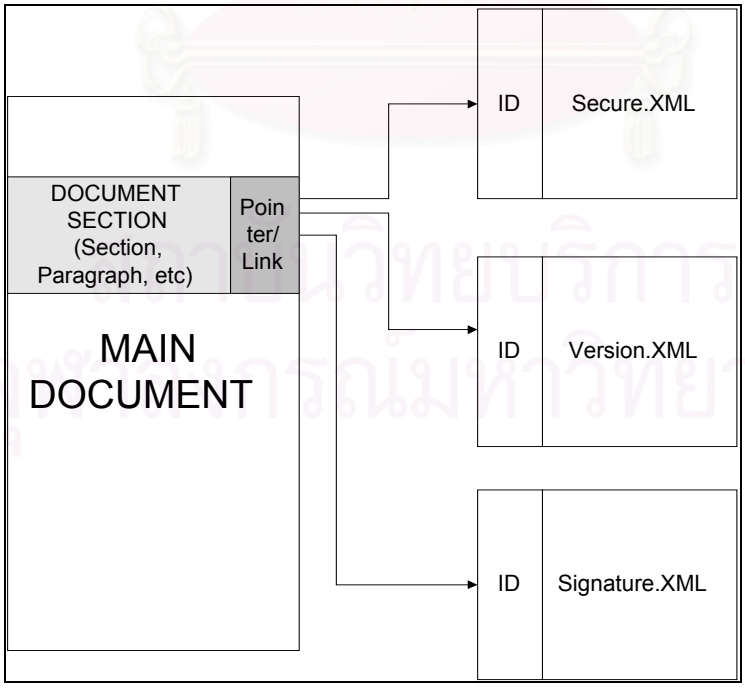
โครงสร้างของเอกสาร

บทนี้อธิบายถึงการออกแบบโครงสร้างของเอกสารที่มีการลงนาม โดยกล่าวถึงลักษณะโครงสร้างโดยทั่วไปของเอกสารลงนามที่ควรมี การออกแบบคำนึงถึงความต้องการของผู้ใช้และผู้พัฒนาระบบที่มีการวิเคราะห์ในบทที่ผ่านมาเป็นหลัก

จากบทที่ผ่านมาสามารถวิเคราะห์ความต้องการออกได้เป็น 3 หมวดหลัก คือ

1. ความต้องการทางด้านการลงนาม
2. ความต้องการทางด้านความปลอดภัยของเอกสาร
3. ความต้องการทางด้านการตรวจสอบชุด (version)ของเอกสาร

เนื่องจากโครงสร้างของเอกสารที่ออกแบบต้องสามารถนำไปใช้ได้กับเอกสารอื่นๆ ที่มีอยู่ในปัจจุบัน ในรูปแบบต่างๆ ดังนั้นจึงได้เสนอการแยกโครงสร้างเอกสารออกเป็นส่วนๆ ตามลักษณะความต้องการ กล่าวคือ โครงสร้างเอกสารที่ระบุความต้องการทางด้านการลงนาม ความต้องการทางด้านความปลอดภัย ของเอกสาร และการตรวจสอบชุด (version)ของเอกสาร โดยในเอกสารหลักมีการใช้ตัวเชื่อมโยง/ ตัวชี้ (Link / Pointer) ไปยังเอกสารย่อยที่กำหนดคุณลักษณะทางด้านต่างๆ ของเอกสารหลัก ดังรูป



รูปที่ 4.1: **แสดงโครงสร้างรวมของเอกสาร**

จากรูป เห็นได้ว่ารูปแบบเอกสารที่มีการลงนามไม่ได้ประกอบด้วยเอกสารฉบับเดียว แต่ยังมีเอกสารย่อยประกอบอีก 3 เอกสาร ซึ่งในอนาคตสามารถเพิ่มขึ้นได้อีกตามความต้องการที่เพิ่มขึ้น โดยเรียกเอกสารทั้งหมดที่ประกอบกันนี้ว่าเป็น **ชุดเอกสารที่มีการลงนาม** (Document Set for Signed Document) โดยเอกสารเสริมทั้ง 3 ฉบับเป็นตัวกำหนดพฤติกรรม (Behavior) และ คุณลักษณะ (Attribute) ในส่วนต่างๆของเอกสารหลัก โดยในแต่ละส่วนของเอกสารหลักมีส่วนของตัวชี้ (pointer) ส่วนในตัวของเอกสารย่อยต้องมีส่วนของการกำหนดเอกสารในแต่ละส่วน (ID) เพื่อให้ตัวชี้สามารถชี้ส่วนของเอกสารหลักได้ถูกต้อง

เอกสารเสริมทั้ง 3 ฉบับอาจสามารถเสริมเอกสารหลักได้มากกว่าหนึ่งเอกสารหลัก ในกรณีที่เอกสารหลักเป็นเอกสารประเภทเดียวกัน ชนิดเดียวกัน ที่มีมากกว่า 1 ชุด สามารถใช้เอกสารเสริมร่วมกันได้ แต่จุดสำคัญคือเอกสารหลักต้องสามารถชี้ไปยังเอกสารย่อยในส่วนต่างๆ ได้ เพื่อกำหนดคุณลักษณะ (Attribute)ของส่วนต่างๆ ในเอกสารหลัก รายละเอียดของการชี้และตัวชี้จะพูดถึงในส่วนของการออกแบบโครงสร้างเอกสารแต่ละส่วน

โครงสร้างของชุดเอกสารที่มีการลงนามในแต่ละส่วนดังต่อไปนี้

- เอกสารหลัก
- ตัวเชื่อมต่อระหว่างเอกสาร
- เอกสารกำหนดสิทธิ์ผู้ใช้และกำหนดการใช้งานข้อมูล
- เอกสารกำหนดชุด (version)ของเอกสาร
- เอกสารกำหนดการลงนามอิเล็กทรอนิกส์

4.1 **โครงสร้างของเอกสารหลัก**

เอกสารหลักคือเอกสารที่กำหนดโครงสร้างของเอกสารที่มีการลงนาม ซึ่งสามารถอยู่ในรูปแบบต่างๆ กัน เช่น ไมโครซอฟท์เวิร์ด เอกซ์เซล เอกซ์เอ็มแอล เอชทีเอ็มแอล และอื่นๆ ซึ่งโครงสร้างเอกสารบางประเภทอาจไม่เหมาะในกับเอกสารที่มีลายเซ็น เนื่องจากโครงสร้างของเอกสารเดิมไม่ได้อำนวยในการอ้างอิงถึงส่วนต่างๆ ของเอกสารได้ละเอียดพอ เช่น การแบ่งเอกสารเป็นย่อหน้า แถว คอลัมน์ ต่างๆ โดยโครงสร้างเอกสารหลักต้องเอื้ออำนวยในการอ้างอิงถึงส่วนต่างๆ ของเอกสารเพื่อให้สามารถกำหนดคุณลักษณะของเอกสารในแต่ละส่วน โครงสร้างที่จำเป็นต่อการสร้างเอกสารหลักสามารถแบ่งย่อยได้ดังนี้

|---เอกสารทั้งเอกสาร (Document)

|---- ส่วนของเอกสาร (Section)

|--- ย่อหน้าหรือฟอร์มของเอกสาร Paragraph (Form)

|--- แถวของเอกสาร (Row)

|--- เขตข้อมูล (Field)

หน่วยย่อยสุดของเอกสารคือเขตข้อมูล (field) ซึ่งแสดงส่วนที่เล็กที่สุดที่มีความหมายในเอกสาร เช่น เขตข้อมูลผู้รับรอง เขตข้อมูลราคา ในแต่ละเขตข้อมูลประกอบกันเป็นแถวของข้อมูล (row) ซึ่งแสดงถึงความสัมพันธ์ของแต่ละเขตข้อมูล เช่น แถวของข้อมูลการสั่งซื้อสินค้าอันประกอบด้วย ราคา รายการสินค้า จำนวน สามารถจัดกลุ่มของแถวที่เกี่ยวเนื่องกันเป็นหนึ่งย่อหน้า (paragraph) เช่น รายการตรวจรับพัสดุ ประกอบด้วยหลายแถวของรายการสินค้าต่างๆ มารวมกันจากส่วนของย่อหน้า สามารถจัดกลุ่มของย่อหน้าที่เกี่ยวข้องเป็นส่วนของเอกสาร (section) เช่นส่วนการขอซื้อในใบขอซื้อ อันประกอบด้วยย่อหน้าชื่อของผู้ซื้อและย่อหน้ารายการของที่ขอจัดซื้อ แต่ที่สำคัญคือในแต่ละส่วนของ เอกสารหลักต้องสามารถสร้างตัวแบ่งแยกส่วนในเอกสารได้ เช่น ในเอกสารเอกซ์เอ็มแอลสามารถใช้แทก (tag) เป็นตัวบอกส่วนต่างๆ ในเอกสารได้

การที่เอกสารหลักสามารถมีตัวแบ่งแยกได้เพื่อเอื้ออำนวยการอ้างอิงในการลงนามทางอิเล็กทรอนิกส์และการตรวจสอบสิทธิ์ของผู้ใช้ได้ในแต่ละส่วนตลอดจนการหาชุด (version)ของเอกสาร

กล่าวโดยสรุปคือเอกสารหลักต้องมีความสามารถในการแบ่งแยกเอกสารโดยมีตัวแบ่งแยกเป็นตัวชี้โดยมีโครงสร้างรวมของเอกสารเป็น

1. เอกสารทั้งหมด (Document)
2. ส่วนของเอกสาร (Section)
3. ย่อหน้า (Paragraph)
4. แถว (Row)
5. เขตข้อมูล (Field)

ตัวอย่างของเอกสารเช่น ในเอกซ์เอ็มแอลสามารถใช้แทก (tag) เป็นตัวกำหนดส่วนต่างๆ ของเอกสารได้ เอกสารหลักที่นำมาใช้ในการลงนามได้ต้องเป็นเอกสารที่มีลักษณะเป็นเนื้อหา (Content Document) คือเอกสารที่เก็บอยู่ในรูปของเนื้อหาข้อมูลมากกว่าเอกสารที่จัดเก็บอยู่ในรูปแบบของ การแสดงผล (Presentation) เช่น เอกสารเอชทีเอ็มแอล (เอชทีเอ็มแอล)

การแปลงเอกสารจากรูปแบบอื่นให้มาอยู่ในเอกสารหลักที่ใช้ในการลงนามต้องคำนึงถึงคุณสมบัติหลักของเอกสารหลักที่กล่าวมาข้างต้นแล้วว่าสามารถแปลงมาได้ยากง่ายเพียงใดหรือต้องสร้างเอกสารใหม่

4.2 โครงสร้างตัวเชื่อม/ ตัวชี้ถึงส่วนของเอกสาร (Link/Pointer)

ส่วนของการอ้างอิงอยู่ในเอกสารหลักที่ใช้ชี้ไปยังเอกสารย่อยเพื่อกำหนดคุณลักษณะของเอกสาร หลัก เช่น เอกสารกำหนดการลงนามอิเล็กทรอนิกส์ กำหนดชุด (version)ของเอกสารและเอกสารกำหนดสิทธิ์ของผู้ใช้ ตัวอ้างอิงในส่วนนี้เป็นตัวกำหนดระบุถึงส่วนต่างๆของเอกสารหลัก โดยตัวชี้สามารถแบ่งได้เป็น 2 ส่วนคือ

1. ฝั่งที่อยู่ด้านเอกสารหลัก ทำหน้าที่เป็นตัวชี้ไปยังเอกสารเสริมต่างๆ
2. ฝั่งที่อยู่ด้านเอกสารเสริม ซึ่งเป็นตัวอ้างอิง (ID) ในการชี้จากเอกสารหลัก

4.2.1 ในส่วนของตัวชี้ (Document Pointer) ที่อยู่ด้านเอกสารหลัก จะมีโครงสร้างของตัวชี้ ดังต่อไปนี้

- ตัวบอกชี้ไปที่เอกสารใดเพื่อระบุเอกสารหลักที่ต้องการชี้
- ตัวบอกชี้ไปที่ส่วนใดของเอกสาร
- ในส่วนประกอบย่อยของเอกสารหลักอาจมีตัวชี้ได้มากกว่าหนึ่ง เช่น ระดับย่อหน้าชี้ไปที่เดียวกันกับในระดับแถว ในกรณีนี้เงื่อนไขที่ทำเฉพาะเจาะจงกว่าจะมีผลบังคับใช้ เช่น เงื่อนไขที่ 1 ทุกแถวในย่อหน้า A เงื่อนไขที่ 2 สำหรับแถวที่ 2 ในย่อหน้า A กรณีเช่นนี้ แถวที่ 2 จะถูกบังคับใช้โดยเงื่อนไขที่ 2
- การกำหนดสถานะของกลุ่มข้อมูลที่ชี้ เช่น มีข้อมูลป้อนเรียบร้อยแล้ว

สามารถสรุปโครงสร้างของตัวชี้ในเอกสารหลักได้ดังรูปที่ 4.2

Type=Document#ReferenceID#status …

Type = ชนิดของเอกสารหลัก อันได้แก่ SECURE | SIGNATURE | VERSION

SECURE = เอกสารกำหนดสิทธิ์ผู้ใช้

SIGNATURE = เอกสารกำหนดรายละเอียดการลงลายเซ็น

VERSION = เอกสารกำหนดชุด (version)ของเอกสาร

รูปที่ 4.1: โครงสร้างของตัวชี้ในเอกสารหลัก

Document = ชื่อของแฟ้มข้อมูลเอกสารเสริมที่อ้างอิงแบบ Absolute หรือ Relative จากสารบบ (Directory) ปัจจุบัน

ReferenceID = เลขตัวอ้างอิงในเอกสารเสริม (ID)

Status = สถานะของตัวชี้เช่น ENABLE | DISABLE | CHECKED | WAIT

ENABLE = ตัวชี้สามารถใช้ได้

DISABLE = ตัวชี้ไม่สามารถใช้ได้

CHECKED = คุณสมบัติในเอกสารเสริมที่ชี้ถูกตรวจสอบแล้ว หมายถึงข้อมูลในเอกสารหลักได้รับการเปลี่ยนแปลงตาม คุณลักษณะที่กำหนดในเอกสารเสริม

WAIT = คุณสมบัติในเอกสารเสริมที่ชี้รอการตรวจสอบ

โดยที่ตัวชี้สามารถมีได้มากกว่าหนึ่ง คั่นแต่ละตัวด้วยช่องว่างและ
สามารถที่จะฝังตัวชี้ในส่วนของตัวกำหนดส่วนของเอกสารได้ดังตัวอย่างต่อไปนี้

<PARAGRAPH SECURE=*Document#ReferenceID.status*, SIGNATURE=*Document#ReferenceID.status*>

.....

.....

</PARAGRAPH>

รูปที่ 4.2: โครงสร้างของตัวชี้ในเอกสารหลัก (ต่อ)

4.2.2 ฝั่งที่อยู่ด้านเอกสารเสริม ซึ่งเป็นตัวอ้างอิง (ID) ในการชี้จากเอกสารหลัก

ในส่วนนี้จะอยู่ในส่วนของเอกสารเสริม โดยใช้เป็นตัวอ้างอิงจากเอกสารหลัก โดยที่เอกสารหลักอาจมีได้มากกว่าหนึ่งที่ชี้มาที่ส่วนของเอกสารเสริม เช่น แถวที่ 1 และ 2 ของเอกสารหลัก มีการกำหนดสิทธิ์ผู้ใช้เหมือนกัน โดยมีรูปแบบของตัวอ้างอิงดังรูปที่ 4.3

ID=*cccccccc*

ccccccc คือ ตัวอักษรชื่อของตัวอ้างอิง / ตัวเลขอ้างอิง

โดยที่ชื่อจะเป็นตัวเลขหรือตัวหนังสือก็ได้แต่ต้องไม่มีช่องว่างขั้น ตัวแปร ID สามารถแทรกในโครงสร้างของเอกสารเสริมได้เช่น

<SIGNATURE ID=12>
……
……
</SIGNATURE>

รูปที่ 4.1: รูปแบบของตัวอ้างอิง

4.3 **โครงสร้างเอกสารกำหนดสิทธิ์ของผู้ใช้และกำหนดการใช้งานของข้อมูล** (Security)

จากความต้องการของผู้ใช้ในบทที่ผ่านมาสามารถสรุปความต้องการที่เกี่ยวข้องกับสิทธิ์ผู้ใช้ได้ดังนี้

ตารางที่ 4.1: แสดงความต้องการด้านความปลอดภัย

ความต้องการด้านความปลอดภัย (Security)	มีการกำหนดกลุ่มของผู้มีสิทธิ์ในลักษณะของเอซีแอล (ACL หรือ Access Control List) สำหรับแต่ละเขตข้อมูลในแบบฟอร์ม
	กำหนดสิทธิ์ของผู้มีลายเซ็นและอำนาจในการแก้ไขเอกสาร (Security Policy)
	กำหนดสิทธิ์ในการแสดงข้อมูลว่าใครมีสิทธิ์ดูได้บ้างโดยขึ้นอยู่กับแบบของเอกสาร เช่น ใบแจ้งเงินเดือน อาจแสดงเงินเดือนให้ดูได้เฉพาะเจ้าของและหัวหน้า แต่ไม่แสดงให้ดูสำหรับผู้ที่ไม่มีสิทธิ์
	การกำหนดอำนาจตามบทบาท (Role) และอำนาจตามบุคคล
	อำนาจการลงนามขึ้นกับเวลา และบุคคลที่ลงนามมีการเปลี่ยนแปลงไป เช่น กรรมการตรวจสอบเฉพาะกิจ มีอำนาจ 6 เดือน
	การกำหนดการเซ็นแทนในช่วงเวลา เช่นผู้มีอำนาจไม่อยู่ และมอบอำนาจให้ผู้อื่นเซ็นแทน
	กำหนดกฎของอำนาจ (Rule of Authorize) ในการลงนาม เช่นผู้มีอำนาจตามวงเงิน

จากความต้องการข้างต้นสามารถนำมาออกแบบโครงสร้างของเอกสารที่กำหนดสิทธิ์ของผู้ใช้ โดยสามารถแบ่งแยกเอกสารได้เป็นส่วนๆ โดยแต่ละส่วนถูกใช้อ้างอิงจากเอกสารหลักในแต่ละส่วนย่อย ตั้งแต่ระดับส่วน (Section) จนถึงระดับเขตข้อมูล (Field) ขึ้นอยู่ว่าการกำหนดสิทธิ์มีรายละเอียดมากเพียงใด โดยส่วนของเอกสารกำหนดสิทธิ์ผู้ใช้แบ่งเป็นสองส่วนใหญ่ คือ

1. ส่วนกำหนดกลุ่มของผู้มีสิทธิ์ใช้ในลักษณะกลุ่ม
2. ส่วนกำหนดสิทธิ์ของผู้ใช้ ที่อ้างอิงจากเอกสารหลัก

1. ส่วนหัวกำหนดกลุ่มผู้ใช้

ส่วนนี้ทำหน้าที่ในการกำหนดกลุ่มผู้ใช้ว่ามีใครบ้างและสิทธิ์ของกลุ่มในการใช้งานเอกสารหลัก โดยมีโครงสร้างดังรูปที่ 4.4

<GROUP NAME=*ccccc* ID=*nnnn* >
<MEMBER FROM= *hh:min:dd:mm:yyyy* TO= *hh:min:dd:mm:yyyy* >
user </MEMBER>
……..
</GROUP>

ccccc	=	ชื่อของกลุ่มผู้ใช้
nnnn	=	เลขอ้างอิง
user	=	ชื่อผู้อยู่ในกลุ่ม
hh	=	เวลาเป็นชม 1-24
min	=	เวลาเป็นนาที 0-59
dd	=	วันที่ 1-31
mm	=	เดือนเป็น 1-12
yyyy	=	ปีเป็น ค.ศ

รูปที่ 4.1: โครงสร้างส่วนหัวกำหนดกลุ่มผู้ใช้

2. ส่วนกำหนดสิทธิ์ของผู้ใช้เพื่อใช้กำหนดในแต่ละส่วนของเอกสารหลักว่าใครมีสิทธิ์มาใช้และแก้ไข โดยชื่อผู้ใช้ สิทธิ์ (Role) อาจได้มาจากส่วนที่หนึ่งของเอกสารที่กำหนดสิทธิ์ในลักษณะกลุ่มหรือ กำหนดโดยตรง โดยรายละเอียดในส่วนที่สอง เป็นตัวกำหนดว่าใคร (Who) มีสิทธิ์ทำอะไร (What) ที่ไหน (Where) เมื่อไร (When) และมีเงื่อนไขอย่างไร (How) โครงสร้างของส่วนที่สองดังรูปที่ 4.6

<SECURE_TAG ID=*nnnn* >
<WHO ROLE = "READ | WRITE | MODIFY | SIGNATURE | ATTACH | COPY"
FROM= *hh:min:dd:mm:yyyy* TO= *hh:min:dd:mm:yyyy* >
<LIST>*ccccc*</LIST>
……..
<GRP>*ggggg*</GRP>
<EXPRESSION>
tag { > | < | ! | =} *alphanum*
</EXPRESSION>
</WHO>
…….
</SECURE_TAG>

nnnn	เลขอ้างอิงจากเอกสารหลัก
ccccc	ชื่อผู้มีสิทธิ์แก้ไข
ggggg	ชื่อกลุ่มผู้มีสิทธิ์แก้ไข
READ	สิทธิ์การอ่านเอกสาร
WRITE	สำหรับการเขียนสร้างเอกสารใหม่เท่านั้น
MODIFY	การแก้ไขเอกสาร
SIGNATURE	สิทธิ์การลงนามเอกสาร
ATTACH	สิทธิ์การแนบเอกสาร
COPY	สิทธิ์การทำสำเนาเอกสาร
hh	เวลาเป็นชม 1-24
min	เวลาเป็นนาที 0-59
dd	วันที่ 1-31
mm	เดือนเป็น 1-12

รูปที่ 4.2: โครงสร้างส่วนกำหนดสิทธิ์ของผู้ใช้

nnnn	เลขอ้างอิงจากเอกสารหลัก
yyyy	ปีเป็น ค.ศ.
tag	ชื่อแทกของเอกสารหลักที่ต้องมีการเปรียบเทียบ เช่น การทดสอบวงเงิน
alphanum	ตัวหนังสือหรือตัวเลขที่ใช้เปรียบเทียบ
ถ้าไม่ได้ระบุ FROM, TO แสดงว่าไม่มีการกำหนดช่วงเวลาของสิทธิ์การใช้	

รูปที่ 4.5: โครงสร้างส่วนกำหนดสิทธิ์ของผู้ใช้ (ต่อ)

4.4 โครงสร้างเอกสารทางด้านการตรวจสอบชุด (version)ของเอกสาร

ในส่วนของการหาชุด (version)ของเอกสารเพื่อใช้ในการตรวจสอบว่าเอกสารมีการแก้ไขไปโดยใครบ้าง รวมถึงการเซ็นในเอกสารว่ามีใครเซ็นบ้าง และก่อนที่จะมีการแก้ไขข้อความเดิมเป็นอย่างไร ทั้งนี้เพื่อให้สามารถตรวจสอบย้อนหลังได้ เอกสารนี้เป็นตัวบ่งบอกถึงการแก้ไขที่มีในเอกสารหลัก

ในส่วนความต้องการทางด้านชุด (version)ของเอกสารสามารถสรุปได้ดังนี้

ตารางที่ 4.1: แสดงความต้องการด้านการตรวจสอบชุด (version)

ความต้องการด้านการตรวจสอบชุด (version) ของเอกสาร	การเก็บบันทึกเอกสารในแต่ละชุด (version)ก่อนที่มีการเปลี่ยนแปลง
	การเรียกดูเอกสารในแต่ละชุด (version)ก่อนที่มีการแก้ไขหรือลงลายเซ็นตามลำดับ

โครงสร้างเอกสารของการตรวจสอบชุด (version) จะมีลักษณะต่างกับโครงสร้างเอกสารอื่นที่กล่าวมาแล้ว คือ ไม่มีส่วนตัวอ้างอิงเพื่อใช้อ้างอิงจากเอกสารหลักที่ต้องการหาชุด (version)ของเอกสาร เนื่องจากมองการแก้ไขเป็นชุด (version)ของเอกสารทั้งหมดมากกว่าเป็นส่วนๆของเอกสาร ดังมีรายละเอียดโครงสร้างดังรูปที่ 4.6

<VERSION_DOC>
<VERSION REVISION=*number* DATE=*dd:mm:yyyy* AUTHOR=*name* >
<TAG NAME=*name_tag* ACTION = SUGGEST | TEMPORARY | MANDATORY | HASH | SIGNED | CANCELED | DELETED | COPIED | MODIFY | ATTACH >*text* </TAG>
……
</VERSION>
</VERSION_DOC>

number	เลขชุด (version)ของเอกสาร
dd	วันที่ 1-31
mm	เดือนเป็น 1-12
yyyy	ปีเป็น ค.ศ.
name	ชื่อผู้แก้ไข
name_tag	ชื่อ XML TAG ที่แก้ไข
text	ข้อความที่แก้ไข (เป็นข้อความหรือข้อมูลที่ผ่านการเข้ารหัสมาแล้ว)

รูปที่ 4.1: โครงสร้างเอกสารของกาตรวจสอบชุด (version)

จากโครงสร้างข้างต้นเห็นได้ว่าการเก็บข้อความที่แก้ไขจะเก็บข้อความทั้งหมดที่เปลี่ยน แทนที่จะเป็นการคำนวณว่ามีการเพิ่ม ลด ที่ใดบ้าง เนื่องจากข้อความในเอกสารหลักส่วนใหญ่เป็นประโยคสั้นๆ การเก็บข้อความที่เปลี่ยนทั้งหมดจะสะดวกในการหามากกว่าการนำมาคำนวณหาชุด (version)ที่ต้องการ ทั้งนี้พื้นที่ในการเก็บข้อมูลไม่ต่างกันมากแต่สามารถคำนวณได้เร็วกว่า ค่า ACTION เป็นตัวกำหนดว่าข้อความที่เก็บมีสถานะอย่างไรโดยมีค่าดังต่อไปนี้

SUGGEST	ข้อความที่เก็บเป็นข้อความแนะนำค่าหรือค่า Default
TEMPORARY	ข้อความที่เก็บเป็นข้อความชั่วคราว
MANDATORY	ข้อความที่เก็บเป็นข้อความถาวร
HASH	ข้อความที่เก็บเป็นข้อความที่ถูกเข้ารหัสโดยฟังก์ชั่นแฮช (Hash)
SIGNED	ข้อความที่เก็บเป็นข้อความที่ถูกลงนาม
CANCELED	ข้อความที่เก็บเป็นข้อความที่ถูกยกเลิก
DELETED	ข้อความที่เก็บเป็นข้อความที่ถูกลบ
COPIED	**ข้อความที่เก็บเป็นข้อความที่ถูกทำสำเนา**
MODIFY	**ข้อความที่เก็บเป็นข้อความที่ถูกแก้ไข**
ATTACH	**ข้อความที่เก็บเป็นชื่อเอกสารที่ถูกแนบ**

การหาแต่ละชุด (version) ของเอกสารสามารถตรงไปดูยังรายละเอียดที่อยู่ภายใต้ในแต่ละชุด (version) ของเอกสาร โดยข้อความต้องถูกเข้ารหัสลับ และดูได้เฉพาะคนที่มีสิทธิ์และถูกระบุในแฟ้มข้อมูลระบุสิทธิ์ผู้ใช้เท่านั้น ในแต่ละชุด (version) ของเอกสารจะเก็บเฉพาะส่วนต่างของชุด (version)ก่อนเท่านั้น

ในกรณีที่ข้อมูลมีจำนวนมากอาจมีการเก็บข้อมูลที่ผ่านฟังก์ชั่นแฮชมาแล้วเพื่อตรวจสอบความถูกต้องของข้อมูลว่าไม่มีการแก้ไขเปลี่ยนแปลง

4.5 โครงสร้างเอกสารทางด้านการลงนามอิเล็กทรอนิกส์

เอกสารส่วนนี้เป็นส่วนที่สำคัญที่สุดของเอกสารการลงนามชุดนี้ เนื่องจากเป็นตัวกำหนดเกี่ยวกับลายเซ็นดิจิตอลที่ใช้ในระบบเอกสารที่มีการลงนาม

สามารถสรุปความต้องการในส่วนของการลงนามอิเล็กทรอนิกส์ในระบบเอกสารที่เป็นอิเล็กทรอนิกส์ได้ดังตารางที่ 4.3

ตารางที่ 4.1: แสดงความต้องการด้านการลงนาม

ความต้องการของเอกสารที่มีการลงนามคนเดียว	ไม่มีการเปลี่ยนแปลงเอกสารเมื่อมีการลงนามเกิดขึ้น
	สามารถตรวจสอบตัวจริงของลายเซ็นได้ว่าใครเป็นผู้เซ็น
	เมื่อมีการแก้ไขเอกสารสามารถลงลายมือชื่อกำกับเช่นเดียวกับการแก้ไขเอกสารทางกระดาษ
	มีการ ทำตราประทับเอกสาร
	การลงลายเซ็นที่ไม่มีช่องให้ใส่ เช่น รับรองสำเนา
	การรับรองเฉพาะแถว เช่น บัญชีคุมวัสดุ
	การแยกระหว่างเอกสารต้นฉบับ (Read/Write) และเอกสารสำเนา (Read Only)
ความต้องการของเอกสารที่มีการลงนามตั้งแต่ 2 คนขึ้นไป • เอกสารที่มีการลงนามแบบลำดับชั้น	สามารถกำหนดส่วนรับผิดชอบ (Protected Field) ในแต่ละคน ว่าใครมีอำนาจแก้ไขในส่วนใด
	ลายเซ็นคนเดียวอาจมีลำดับการลงนามไม่เหมือนกัน เช่น ผู้ขอเบิกเงินและผู้รับเงิน

ตารางที่ 4.3: แสดงความต้องการด้านการลงนาม (ต่อ)

	มีการกำหนดต้นไม้แห่งสิทธิ์ (Tree of Authorization) ในการกำหนดลำดับของการลงนาม
	ต้องสามารถเปลี่ยนแปลงอำนาจและลำดับการลงนามได้ (Authorization Flow)
ความต้องการทางด้านลายเซ็นดิจิตอล	การตรวจสอบลายเซ็นในเอกสารกับตัวจริงของบุคคลที่ลงนาม (Authentication)

โครงสร้างเอกสารที่มีการลงนามแบ่งเป็น 2 ส่วนหลักคือ

1. ส่วนกำหนดวิธีการลงนามโดยทั่วไป คือการกำหนดเทคนิคและเทคโนโลยีที่ใช้ในการลงนาม ในส่วนนี้เป็นการบอกว่าชุดเอกสารที่ใช้อยู่นี้ใช้เทคโนโลยีในการลงลายเซ็นแบบใด รวมถึงกำหนดคุณลักษณะของลายเซ็นแต่ละคนที่ใช้ตามเทคโนโลยีที่ระบุ เช่น ถ้าเป็นอาร์เอสเอ (RSA) เป็นกุญแจทั่วไป (Public Key) ของผู้ใช้แต่ละคนที่ใช้ในเอกสารชุดนี้ รวมถึงส่วนเสริมต่างๆ เช่น ภาพลายเซ็นในการลงนาม สามารถสรุปโครงสร้างได้ดังนี้

- Algorithm คือ กระบวนการทำงานหรือสูตรการคำนวณ ในการลงนาม เช่น อาร์เอสเอ (RSA) หรือ ดีเอสเอ (DSA)
- Hash-Algorithm คือ กระบวนการทำงานหรือสูตรการคำนวณที่ใช้ในแฮชฟังก์ชั่น (Hash Function) และ เอ็มดีเอส (MDS)
- PERSON-PUBLIC-KEY คือ กุญแจทั่วไป (Public Key) ของผู้ที่มีสิทธิ์ลงนามเพื่อใช้ในการเข้ารหัส ถอดรหัสข้อมูล
- Digital Image คือ ภาพของลายเซ็นแบบดิจิตอล (Digital Signature) ที่แต่ละคนเข้ารหัสไว้) ต้องใช้กุญแจส่วนบุคคล (Private Key) ในการถอดรหัส ในส่วนนี้คือเป็นออปชั่นเพิ่มเติม

2. ส่วนกำหนดการลงลายเซ็น คือ ส่วนที่ใช้กำหนดคุณลักษณะต่างๆ ของการลงลายเซ็น และเก็บรายชื่อผู้ที่ต้องลงนามทั้งหมด โดยสามารถตรวจสอบได้ว่ายังขาดอยู่อีกกี่คน โดยดูได้จาก จำนวนผู้ที่ลงลายเซ็นไปแล้วในเอกสารกำหนดชุด (version) ในส่วนของ AUTHOR ส่วนผู้ที่มีอำนาจในการลงลายเซ็นถูกกำหนดอยู่ในเอกสารกำหนดสิทธิ์ผู้ใช้ (Security) ข้อมูลที่เพิ่มเติม แก้ไข ไม่ได้ถูกเก็บที่เอกสารหลัก (Main) หรือเอกสารทางด้านการลงนามทางอิเล็กทรอนิกส์ แต่ถูกเก็บอยู่ในเอกสารการตรวจสอบชุด (version) (version)

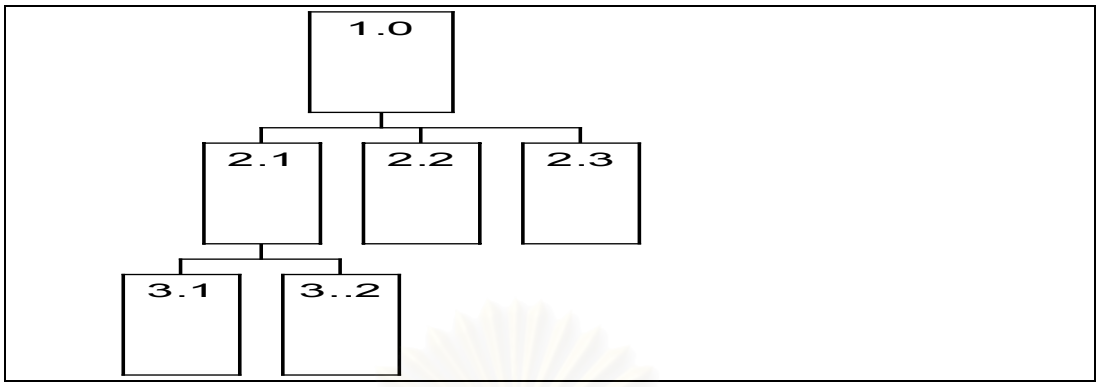
ในส่วนนี้เราจะกำหนดลักษณะของการเซ็นก่อนหลัง โดยใช้บทบาทกับตำแหน่งเป็นหลัก เช่น บัญชี ผู้ตรวจสอบ เนื่องจากจะเป็นอิสระไม่ขึ้นบุคคล ในอนาคตเมื่อมีการเปลี่ยนบุคคลก็สามารถทำได้ง่าย เช่น บัญชีเปลี่ยนจากนาย ก เป็น นาย ข โดยสามารถเปลี่ยนได้จากกลุ่มผู้ใช้ในเอกสารกำหนดสิทธิ์

สามารถสรุปโครงสร้างเอกสารทางด้านการลงนามอิเล็กทรอนิกส์ได้ดังรูปที่ 4.7

```
<SIGNATURE_DOC>
        <SIGNATURE_HEAD  NAME = name>
                <SIGN_ALGORITHM>algorithm<SIGN_ALGORITHM>
                <HASH_ALGORITHM>hash<HASH_ALGORITHM>
                <KEY OWNER=who>
                        <PUBLICE_KEY>
                                key……...
                        </PUBLICE_KEY>
                <SIGNATURE_IMAGE>src_image</SIGNATURE_IMAGE>
                </KEY>
        </SIGNATURE_HEAD>
        <SIGNATURE ID = id
          SIGNTYPE = {COSIGN | HIERARCHY | SINGLE | ACKNOWLEDGE} >
                <SIGNER  TREE=mm.nn> Sign_name</SIGNER>
                <SIGNER  TREE=mm.nn> Sign_name</SIGNER>
        </SIGNATURE>
</SIGNATURE_DOC>
```

algorithm	ชื่อขั้นตอนที่ใช้ในลายเซ็นดิจิตอล
hash	ชื่อขั้นตอนที่ใช้ในกระบวนการทำค่าแฮช
name	ชื่อส่วนหัวของการกำหนดรายละเอียดการเซ็น
key	กุญแจทั่วไป (Public Key) ของ name เป็นตัวอักษร (Text)
sign_name	ชื่อของผู้มีอำนาจเซ็น
src_image	ชื่อแฟ้มข้อมูลของรูปภาพลายเซ็น
id	เลขอ้างอิงจากเอกสารหลัก
who	ชื่อผู้ลงลายเซ็นหรือชื่อกลุ่มผู้มีสิทธิ์ในการลงลายเซ็น
mm.nn	เลขประจำต้นไม้แสดงต้นไม้ของสิทธิ์ว่าใครมีลำดับการลงนามก่อน หลัง

รูปที่ 4.1: โครงสร้างเอกสารทางด้านการลงนามอิเล็กทรอนิกส์



รูปที่ 4.2: ต้นไม้ของลำดับการลงนาม

จากรูปเห็นได้ว่า 1.0 ต้องเซ็นก่อน 2.1 และ 2.1 มีลำดับการลงนามเท่ากับ 2.2 และ 2.3 แต่มาก่อน 3.1 และ 3.2 การกำหนดเลขประจำต้นไม้ลำดับการลงนามเพื่อใช้ในการลงลายเซ็นแบบลำดับชั้นในกรณีที่มีสิทธิ์ในการลงนาม เอกสารเหมือนกัน

4.6 **ความสัมพันธ์ของโครงสร้างเอกสารและความต้องการของผู้ใช้**

หัวข้อนี้จะอธิบายถึงพฤติกรรมในการทำงานของชุดเอกสารที่มีการลงนามอันประกอบด้วย เอกสารหลัก และเอกสารเสริมอีก 3 ตัว อันได้แก่

1. เอกสารกำหนดสิทธิ์ผู้ใช้
2. เอกสารกำหนดการลงนาม
3. เอกสารกำหนดชุด (version)ของเอกสาร

โดยอธิบายในลักษณะของตารางตามตารางที่ 4.4 โดยอิงจากตารางความต้องการของผู้ใช้ในบทที่ 3 เป็นหลัก ส่วนวงจรชีวิตของเอกสารจะกล่าวถึงในบทที่ 5 เรื่อง วงจรชีวิตของชุดเอกสารในระหว่างขบวนการลงนาม ในตารางที่ 4.4 มีการอ้างถึงแทกต่างๆ ที่ใช้ในเอกสารแต่ละเอกสารว่ามีการทำงานสัมพันธ์กันอย่างไร ตารางนี้เป็นตัวกำหนดพฤติกรรมของเอกสารในการออกแบบและพัฒนาระบบเอกสารลงนามต่อไป

ตารางที่ 4.1: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
1.1	แต่ละส่วนของเอกสารหลักที่มีการลงนามมีการคำนวณตัวแทนข้อมูลด้วยกระบวนการหาค่าแฮช (Hash Algorithm) ที่ระบุในส่วนหัวของเอกสารการกำหนดการลงลายเซ็น ตัวแทนของข้อมูลถูกเก็บในเอกสารกำหนดชุด (version)เพื่อบอกว่าเป็นชุด (version)ที่เท่าไรของการแก้เอกสาร โดยที่การแก้แต่ละครั้งมีการเก็บข้อมูลตัวแทนไว้เพื่อคำนวณเปรียบเทียบกับข้อมูลดั้งเดิมว่ามีการแก้ไขหรือไม่ โดยมีการทำงานดังนี้ 1. ส่วนของข้อมูลที่มีการลงนามหลังจากแก้ไขเสร็จแล้วมีการคำนวณตัวแทนข้อมูลโดยใช้ <HASH ALGORITHM> ตามที่ระบุ 2. นำค่าแฮช (HASH) ที่ได้ไปใส่ในเอกสารกำหนดชุด (version) ภายใต้ <VERSION> ของเอกสารส่วนนั้นที่ชี้โดยตัวชี้จากเอกสารหลัก (Pointer)	<VERSION>	<HASH_ALGORITHM> <SIGN_ALGORITHM> <PUBLIC_KEY>	

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
	3. นำค่าแฮช (HASH) ที่ได้ไปเข้ารหัสข้อมูลโดยใช้กุญแจส่วนบุคคล (Private Key) ของผู้ลงนาม ถ้าต้องการดูว่ามีการแก้ไขเอกสารหรือไม่ สามารถคำนวณกลับกันได้ คือ 1. นำค่าแฮช (HASH) มาถอดรหัสโดยใช้ <PUBLIC_KEY> ของผู้ส่ง 2. คำนวณค่าแฮช (HASH) ของเอกสารในส่วนที่ต้องการตรวจสอบ 3. ถ้า 1 และ 2 ตรงกัน แสดงว่าข้อมูลถูกต้องไม่มีการเปลี่ยนแปลงโดยบุคคลอื่น			
1.2	สามารถตรวจสอบตัวจริงของลายเซ็นโดยการเข้ารหัสข้อมูลโดยมีกุญแจส่วนบุคคล (PRIVATE KEY) สามารถใช้กุญแจทั่วไป (PUBLIC KEY) ของคนที่ได้รับอนุญาตเปิดดูได้ แสดงว่าเป็นของเจ้าของกุญแจทั่วไป (PUBLIC KEY) ที่ใช้ถอดรหัสจริง	<VERSION>	<PUBLIC_KEY>	

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
1.3	เมื่อมีการแก้ไขเอกสารสามารถลงนามรับรองการแก้ไขได้ โดยการใช้กุญแจส่วนบุคคล (PRIVATE KEY) เข้ารหัสเอกสารและนำไปเก็บไว้ที่เอกสารกำหนดชุด (version) ถ้าเป็นเอกสารใหม่จะเริ่มที่ ชุด (VERSION) 1.0 จนเมื่อมีการแก้ไขอีกจะเพิ่มชุด (version)เป็น 2.0 หรือใช้กระบวนการหาค่าแฮช (HASH ALGORITHM) ในการคำนวณตัวแทนของข้อมูลเพื่อเพิ่มความเร็วในการลงลายเซ็น และต้องการเปิดให้ดูข้อมูลได้ แต่ไม่ต้องการแก้ไขข้อมูล	<VERSION>	<HASH_ALGORITHM> <PUBLIC_KEY>	<SECURE_TAG>
1.4	การประทับเอกสารมีลักษณะคล้ายกับการลงนาม เพียงแต่เป็นชนิดการลงลายเซ็นที่มีการจัดกลุ่มได้มากกว่า 1 คน หรือกล่าวอีกนัยหนึ่ง คือ ตรายางคือการลงลายเซ็น ที่มีผู้มีสิทธิ์ได้มากกว่าหนึ่งคนในการใช้ตรายาง		<SIGNATURE_HEAD>	<SECURE_TAG>

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
1.5-1.6	มีลักษณะเหมือนการลงลายเซ็นรับรองเฉพาะส่วน โดยสามารถระบุเฉพาะส่วนของเอกสารได้ เช่น เอกสารทั้งเอกสาร ส่วนของเอกสาร ย่อหน้าฟอร์มของเอกสาร หรือการลงนามเฉพาะแถวหรือเขตข้อมูลของเอกสาร โดยตัวชี้ในเอกสารหลักเป็นตัวเชื่อมไปยังเอกสารประกอบต่างๆ เช่น เอกสารกำหนดชุด (version) ของเอกสาร เอกสารกำหนดการลงนาม และเอกสารกำหนดสิทธิ์	<VERSION_TAG>	<SIGNATURE>	<SECURE_TAG>
1.7	สามารถแยกระหว่างเอกสารต้นฉบับเอกสารสำเนาได้โดยการกำหนดสิทธิ์การแก้ไขของเอกสารและคุณลักษณะ (Attribute) ของเอกสารว่าเอกสารตัวนี้เป็นเอกสารต้นฉบับและเอกสารสำเนา	<VERSION>		<SECURE_TAG>
2.1	จากโครงสร้างของตัวชี้เอกสารหลักไปยังเอกสารอื่นอีก 3 เอกสารทำให้สามารถระบุกำหนดส่วนการลงนาม การแก้ไขสิทธิ์ผู้ใช้ ได้อย่างอิสระออกจากกัน จากแต่ละส่วนของเอกสาร	<SECURE> <SIGNATURE>		
2.2-2.3	ลายเซ็นของคนหนึ่งอาจมีผลในแต่ละส่วนของแต่ละเอกสารไม่เหมือนกัน		<SIGNATURE>	<GRP> <SECURE_TAG>

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
	สามารถกำหนดสิทธิ์ของลายเซ็นในเอกสารแต่ละ ส่วนได้ ส่วนการกำหนดลำดับของการลงลายเซ็นนั้นสามารถกำหนดได้จากต้นไม้แห่งการลงนาม (TREE OF AUTHORIZE) ในการกำหนดลำดับของความสำคัญก่อนหลังและชนิดของการลงลายเซ็นว่าเป็นแบบใด แบบลำดับก่อนหลัง เซ็นร่วม เป็นต้น			
3.1	การใช้เทคนิคกุญแจแบบอสมมาตร(Asymmetric Key) ทำให้ต้องมีกุญแจในกรเข้ารหัสข้อมูล 2 ชุด คือ กุญแจทั่วไป (Public Key) และกุญแจส่วนบุคคล (Private Key) ซึ่งกุญแจทั่วไป (Public Key) สามารถแจกจ่ายได้ทั่วไป แต่กุญแจส่วนบุคคล (Private Key) จะเก็บอยู่กับเจ้าของคนเดียว ทำให้สามารถตรวจสอบได้ว่าลายเซ็นในเอกสารเป็นของจริงหรือไม่ โดยดูได้จากการใช้กุญแจส่วนบุคคล (Private Key) ของคนซึ่งเป็นคนถอดรหัส		<SIGNATURE_HEAD>	

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
3.2	สามารถกำหนดกระบวนการทำงาน (Algorithm) ในการทำลายเซ็นแบบดิจิตอล (Digital Signature) ได้จากการกำหนดในแทก <ALGORITHM> และ <HASH_ALGORITHM>		<SIGNATURE_HEAD> <ALGORITHM> <HASH_ALGORITHM>	
4.1	สามารถแนบแฟ้มข้อมูลเอกสารลายเซ็นเพื่อใช้แสดงในเอกสาร โดยแฟ้มข้อมูลลายเซ็นนี้ถูกให้รหัสโดยใช้กุญแจส่วนบุคคล (Private Key) ของเจ้าของ ดังนั้นต้องใช้กุญแจทั่วไป (Public Key) ของเจ้าของลายเซ็นเท่านั้นในการถอดรหัส เพื่อให้ได้ภาพของลายเซ็นที่ถูกต้อง		<SIGNATURE_IMAGE>	
4.2-4.3	ดูรายละเอียดในบทการออกแบบระบบ (System Design)			
5.1	สามารถบันทึกเอกสารส่วนที่มีการเปลี่ยนแปลงในแต่ละชุด (version) ลงในเอกสารความปลอดภัย (Security) โดยเอกสารแต่ละส่วนที่เปลี่ยนแปลงจะ ถูกอ้างอิงไปยังเอกสารกำหนดชุด (version) โดยใช้ตัวชี้ และที่เอกสาร ชุด (VERSION)มีตัว ID เป็นตัวอ้างอิง โดยการเก็บข้อมูลในส่วนนี้ อาจเป็นข้อมูลที่เปลี่ยนแปลงทั้งหมด หรือ ข้อมูลที่ผ่านฟังก์ชั่นแฮช (HASH) เพื่อหาตัวแทนของข้อมูล	<VERSION_TAG>		

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
5.2	สามารถเรียกดูเอกสารในแต่ละชุด (version)จากข้อมูลที่อยู่ใต้ <VERSION> TAG ที่อ้างอิงจากเอกสารหลักโดยใช้ตัวชี้ ไปยังเอกสารชุด (version) (ID) ภายใต้ <VERSION_TAG>	<VERSION> <VERSION_TAG>		
6.1-6.2	จาก <SIGNATURE> TAG ทำให้สามารถหาได้ว่าเอกสาร ในส่วนที่ระบุ ต้องการการลงนามแบบใด (เซ็นร่วม ตามลำดับชั้น) ถ้าเป็นแบบเซ็นร่วม (ลำดับการลงลายเซ็นไม่สำคัญ) สามารถตรวจสอบได้จาก	<VERSION>	<SIGNATURE>	<GROUP> <MEMBER>

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
	<SIGNATURE> แทกว่าใครบ้างมีสิทธิ์ลงนามในส่วนนี้ จากนั้นจะตรวจสอบไปยังเอกสารชุด (version) ว่าคนที่มีชื่ออยู่ลงนามหรือยัง โดยดูได้จาก <VERSION> แทก ตัว AUTHOR จะสามารถตรวจสอบได้ว่าใครบ้างลงนามไปแล้ว (ก่อน / หลัง ตามเลขที่ชุด (version)) โดยผู้ลงนามใหม่จะเข้ารหัส (Encrypt) ข้อมูลของคนเดิม ในทางกลับกันถ้าต้องการดูข้อมูลต้อง ถอดรหัส(Decrypt)ข้อมูลกลับกับตอนที่ข้อมูล ถ้าเป็นการลงนามตามลำดับชั้นก่อนมีการลงนามต้องตรวจสอบไปที่ต้นไม้ลำดับการลงนาม (TREE OF AUTHORIZE) ว่าใครมีลำดับการลงลายเซ็นก่อนหลังตามลำดับ จากข้อมูลที่เก็บในแทก<SIGNER>			

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
7.1	ในส่วนต่างๆ ของเอกสารหลักสามารถชี้ไปยังเอกสารที่กำหนดสิทธิ์ผู้ใช้ โดยในส่วนของการกำหนดสิทธิ์ผู้ใช้สามารถกำหนดกลุ่มของผู้ใช้ที่มีสิทธิ์เหมือนกัน ทำให้สะดวกในการกำหนดสิทธิ์ในลักษณะเป็นกลุ่ม ทำให้การเพิ่มจำนวนผู้มีสิทธิ์ทำได้ง่ายกว่าการมอบหมายสิทธิ์ให้แต่ละคน			<GROUP>
7.2 7.3	ในเอกสารกำหนดสิทธิ์ <SECURE_TAG> <WHO> สามารถกำหนดได้ว่าใครมีสิทธิ์ในการลงลายเซ็นหรือแก้ไขเอกสาร รวมทั้งสิทธิ์ในการดูเอกสาร เช่น อ่าน (READ)			<SECURE_TAG> <WHO>
7.4	การใช้ <GRP> แทก ทำให้สามารถกำหนดอำนาจตาม Role ได้ โดยให้ชื่อของ <GRP> TAG คือชื่อของบทบาท (Role) และกำหนดสมาชิกที่อยู่ในบทบาท (Role) ได้ โดยใน <GROUP> TAG สามารถกำหนดสิทธิ์ของผู้ใช้ได้ว่ามีอะไรบ้าง			<GRP>

ตารางที่ 4.4: แสดงความสัมพันธ์ของความต้องการและโครงสร้างของเอกสาร (ต่อ)

ตารางความต้องการ	พฤติกรรมของชุดเอกสารที่มีการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดชุด (version)	แทก (TAG) ที่ใช้ในเอกสารกำหนดการลงนาม	แทก (TAG) ที่ใช้ในเอกสารกำหนดสิทธิ์ผู้ใช้
7.5-7.6	ใน <SECURE_TAG> สามารถกำหนดผู้มีอำนาจลงนาม โดยสามารถกำหนดช่วงเวลาที่มีสิทธิ์ได้ว่าอยู่ในช่วงใด จาก Attribute FROM และ TO			<SECURE_TAG> <WHO>
7.7	ใน TAG <WHO> สามารถกำหนดนิพจน์ทางคณิตศาสตร์โดยใช้ <EXPRESSION> TAG ในการกำหนดนิพจน์ทางคณิตศาสตร์เพื่อใช้ในการหาเลื่อนไขในการกำหนดสิทธิ์ได้ เช่น ตัวเลขของเงินมากกว่าที่กำหนดจะไม่สามารถลงนามได้			<WHO> <EXPRESSION>
8.1-8.2	ในเอกสารหลักสามารถระบุเอกสารที่แนบได้โดยระบุใน <ATTACH> TAG <ATTACH> อยู่ภายใต้ส่วนของเอกสารนั้น แต่การเพิ่ม <ATTACH> TAG ได้นั้นโปรแกรมต้องตรวจสอบสิทธิ์จากเอกสารกำหนดสิทธิ์ก่อนว่ามีสิทธิ์ในการแนบเอกสารหรือไม่ ถ้าไม่จะไม่สามารถแนบเข้าในเอกสารหลักได้			<SECURE_TAG>

บทที่ 5

วงจรชีวิตของชุดเอกสารในระหว่างกระบวนการลงนาม

วงจรชีวิตของชุดเอกสารที่มีการลงนามในแง่ผู้ใช้สามารถแบ่งออกได้เป็นช่วงต่างๆได้ดังนี้

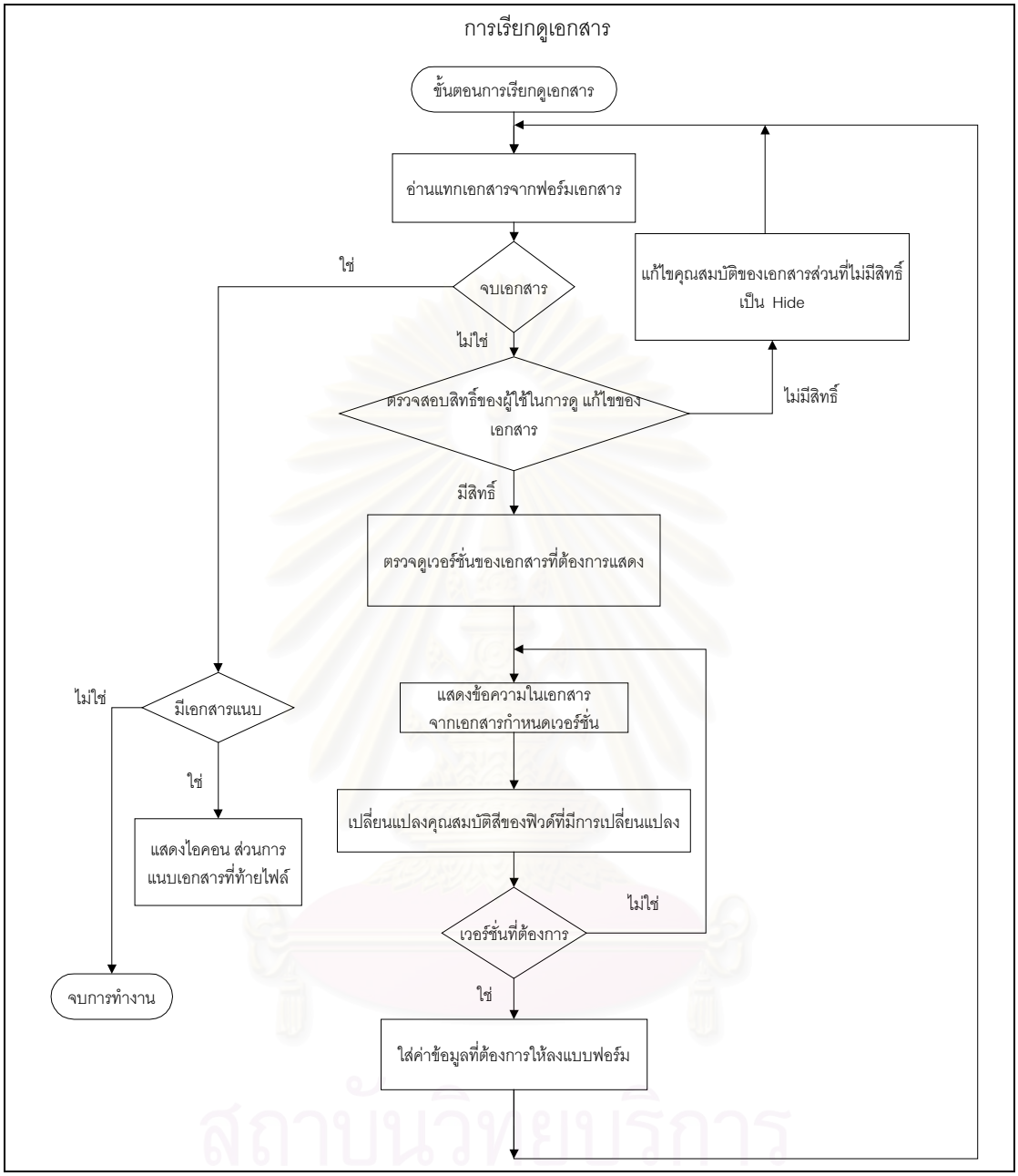
1. การสร้างเอกสาร
2. การเรียกดูเอกสาร
3. การแก้ไขเอกสาร
4. การลงนามเอกสาร
5. การแนบเอกสาร
6. การส่ง/รับเอกสาร

1. การสร้างเอกสารจะถูกกล่าวถึงโดยละเอียดในบทที่ 8

2. การเรียกดูเอกสาร ในการเรียกดูเอกสารนั้นต้องสามารถป้องกัน ผู้ไม่มีสิทธิ์ดู รวมถึงความสามารถในการเรียกดูแต่ละชุด (version)ของเอกสารทั้งก่อนและหลังการแก้ไขโดยมี หลักการทำงาน ดังนี้

2.1. ตรวจสอบผู้ใช้ระบบกับเอกสารกำหนดสิทธิ์ว่าผู้ใช้ระบบปัจจุบันมีสิทธิ์ในการดูเอกสารหรือไม่ ถ้าไม่มีสิทธิ์ต้องยกเลิกการแสดงในส่วนนั้น โปรดจำไว้ว่าสามารถแยกเอกสารหลักเป็นส่วนๆ ได้

2.2. ถ้ามีสิทธิ์ดู ผู้ใช้สามารถเรียกได้ว่าจะดูชุด (version)ใดโดยการนำ ข้อมูลในแต่ละชุด (version)มาแสดง อาจใช้สีเป็นตัวแยกในแต่ละชุด (version) การเรียกดูเอกสาร ในแต่ละชุด (version)ทำให้สามารถทราบได้ว่าเอกสารมีการเปลี่ยนแปลงไปอย่างไร โดยใครบ้าง โดยไม่สามารถย้อนไปแก้ไขในชุด (version)เก่าได้ เพื่อการเก็บประวัติของเอกสารที่ถูกต้องตรงความเป็นจริง



รูปที่ 5.1: ขั้นตอนการเรียกดูเอกสาร

3. การแก้ไขเอกสาร ในการแก้ไขเอกสารที่สร้างขึ้นมาแล้วสามารถแยกการแก้ไขให้ละเอียด ลงไปเป็น การแก้ไขส่วนต่างๆ ของเอกสาร การแก้ไขข้อความในเอกสาร การลงนามรับรองเอกสารและการแก้ไขใหม่ในแต่ละส่วนของเอกสาร โดยเน้นไปที่การแก้ไขเอกสารหลักแต่ต้องปรับปรุงเอกสารอื่นเพื่อให้ข้อมูลมีความสัมพันธ์กัน

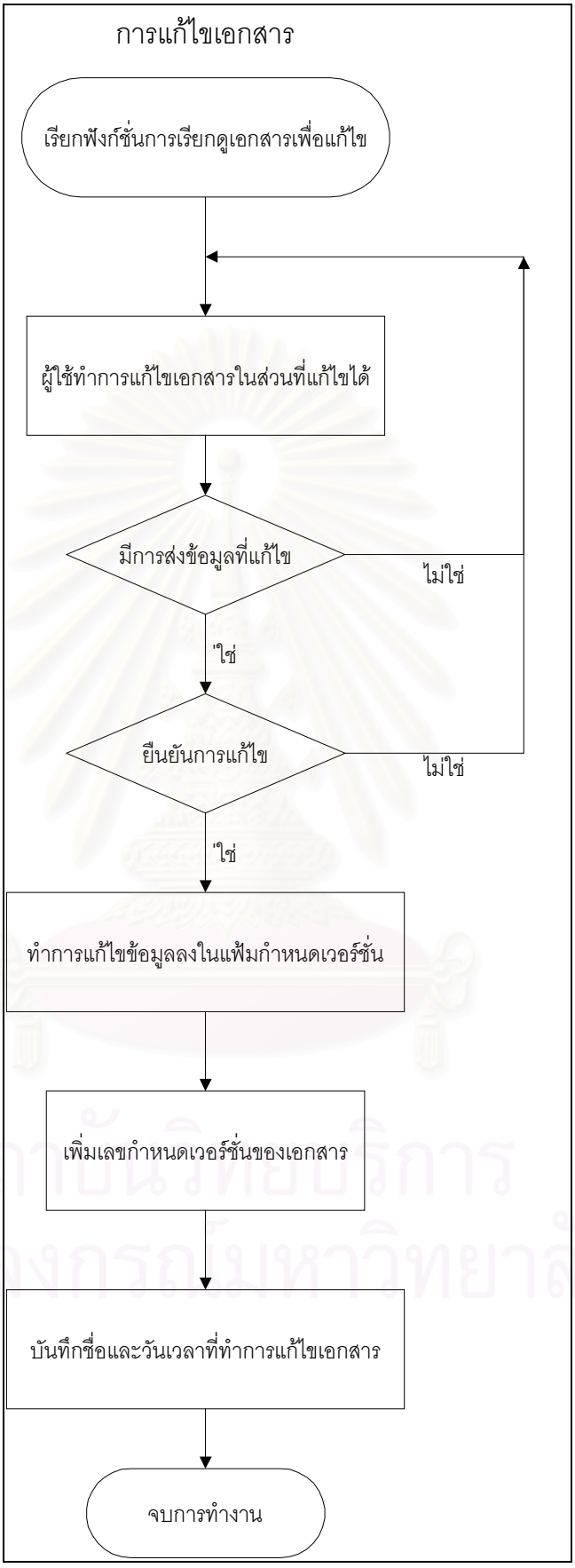
การแก้ไขข้อความในแต่ละส่วนของเอกสาร โดยในแต่ละการแก้ไขต้องมีการเก็บข้อความเดิมก่อนที่มีการแก้ไขไว้เสมอเพื่อการเรียกดูในภายหลัง โดยขั้นตอนในการแก้ไขเอกสารมีดังต่อไปนี้

3.1. ตรวจสอบชื่อผู้ใช้ระบบกับเอกสารกำหนดสิทธิ์ในส่วนที่ต้องการแก้ไข โดยใช้ตัวชี้ (Pointer)ในเอกสารหลักเป็นตัวชี้ไปยังเอกสารกำหนดสิทธิ์ เพื่อดูว่าผู้ใช้ระบบมีสิทธิ์แก้ไขระบบ หรือไม่ โดยดูได้จากชื่อใน <LIST> หรือชื่อสมาชิกในกลุ่ม <GRP> ในกรณีที่สิทธิ์เป็นชื่อกลุ่มแทนที่จะเป็นตัวบุคคล

3.2. ถ้าตรวจสอบแล้วมีสิทธิ์จะอนุญาตให้ผู้ใช้ระบบทำการแก้ไขข้อมูลในส่วนนั้นๆ โดยต้องมีการเก็บข้อมูลเก่าไว้ก่อน โดยเก็บใน <VERSION> TAG ซึ่งมีรายละเอียดของวันที่ ผู้แก้ไข และข้อความที่แก้ไข ในลักษณะของข้อความที่แก้ไขจะมีการเข้ารหัสกุญแจส่วนบุคคล (PRIVATE KEY) ของผู้ใช้ระบบเพื่อให้สามารถตรวจสอบได้ว่าใครเป็นผู้แก้ไข

3.3. ใส่กุญแจทั่วไป (PUBLIC KEY) ของผู้แก้ไขในเอกสารการลงนามเพื่อ ประโยชน์ในการใช้ในการถอดรหัสและตรวจสอบเอกสาร

หลังการสร้างเอกสารหลักแล้วไม่สามารถเพิ่ม / ลด ส่วนต่างๆในเอกสารหลักได้จากระบบ ถ้าต้องการเพิ่มเติมแก้ไขต้องใช้โปรแกรมสร้างเอกสาร (EDITOR) ในการแก้ไขเหมือนการสร้างเอกสารใหม่



รูปที่ 5.2: ขั้นตอนการแก้ไขเอกสาร

4. การลงนามเอกสารถือเป็นส่วนสำคัญที่สุดในระบบนี้ การลงนามอาจครอบคลุมได้หลายส่วน เช่น การลงนามทั้งเอกสาร เฉพาะส่วน เฉพาะแถว ทั้งนี้ขึ้นอยู่กับตัวชี้ในเอกสารหลักว่าตัวชี้นั้นครอบคลุมในส่วนใดของเอกสาร โดยการลงนามมีหลักการทำงานดังนี้

4.1. ตรวจสอบผู้มีอำนาจในการลงนามในเอกสารกำหนดการลงนามว่ามีสิทธิ์ในการลงนามหรือไม่ อยู่ในช่วงเวลาที่กำหนดหรือไม่ ผู้มีอำนาจในการลงนามและผู้มีอำนาจแก้ไขอาจเป็นคนเดียวกันหรือคนละคนก็ได้ เช่น ตัวเลขอาจมีเพียงพนักงานบัญชีมีสิทธิ์แก้ไข แต่ผู้จัดการมีอำนาจในการลงนามรับรอง

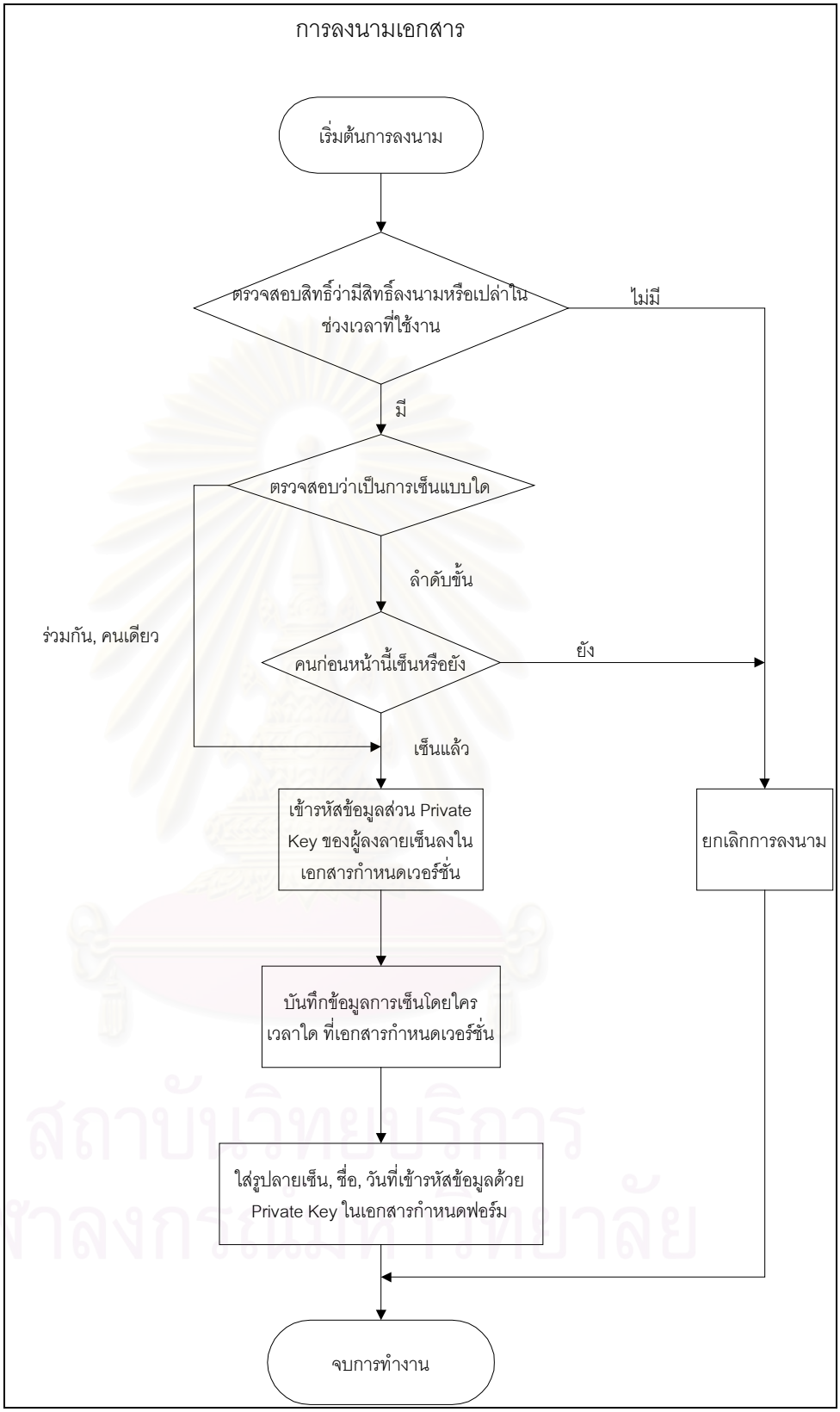
4.2. ถ้ามีอำนาจในการลงนามให้ใช้กุญแจส่วนบุคคล (PRIVATE KEY) ของผู้นั้นในการเข้ารหัสข้อมูลต่างๆ ในแต่ละฟิลด์ที่อยู่ภายใต้ส่วนของเอกสารที่การลงนามครอบคลุม พร้อมทั้งเก็บในเอกสารกำหนดชุด (version)และฟังก์ชั่นผู้ที่ลงนาม เวลาที่ต้องลงนาม

ในการลงนามนี้สามารถกำหนดขั้นตอนวิธีที่ใช้ในการลงนามต่างกันได้ เหตุผลเพื่อรองรับเทคโนโลยีในอนาคต อีกเหตุผลหนึ่ง คือสามารถกำหนดขั้นตอนวิธีที่แตกต่างกันในการลงนามแต่ละส่วน ดังนั้นอาจกำหนดให้ขั้นตอนวิธีง่ายสำหรับเอกสารที่ต้องการเซ็นเพื่อรับรู้ ทำให้ง่ายและรวดเร็วต่อการคำนวณผลลัพธ์

ในกรณีที่เป็นการเซ็นร่วมหรือลำดับชั้นที่มากกว่าหนึ่งคนมีขั้นตอนเพิ่มเติมดังนี้คือ

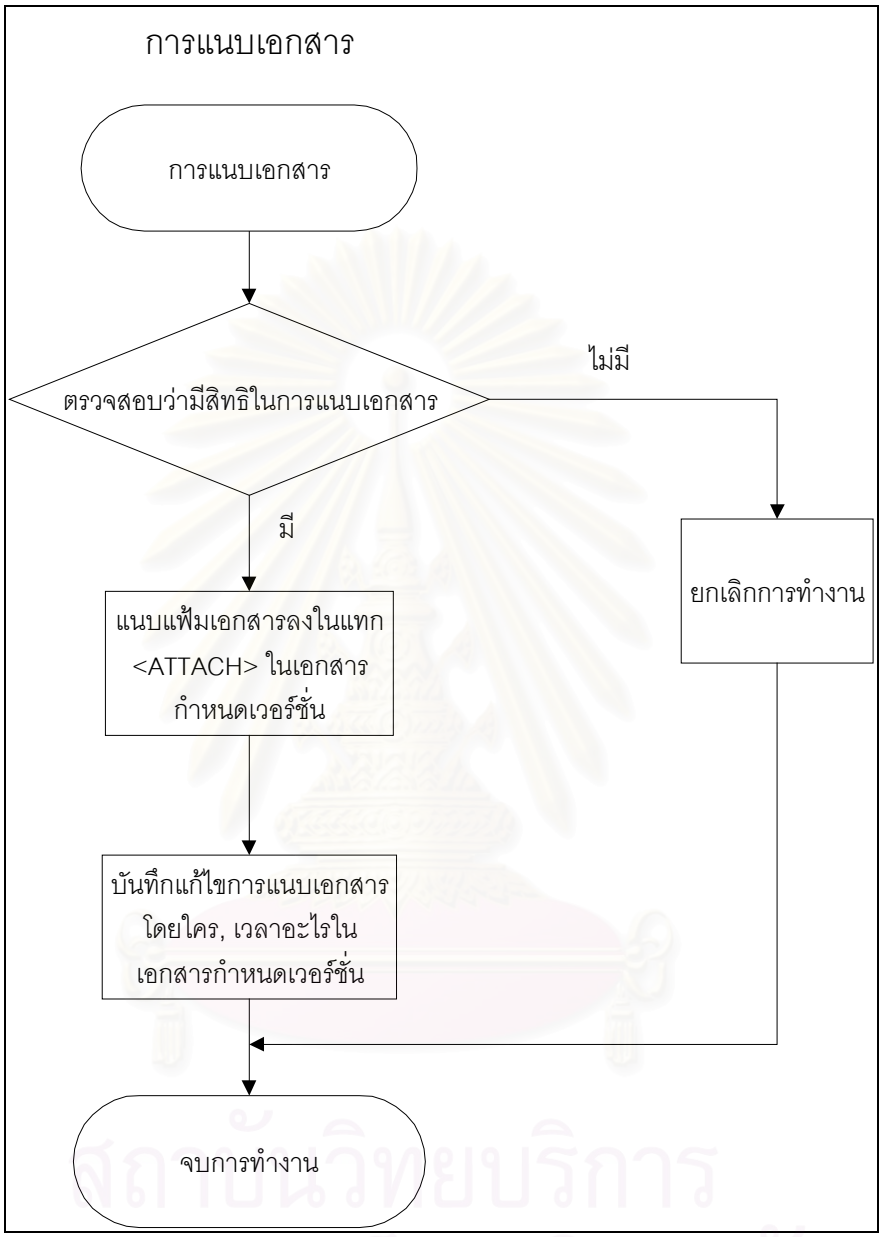
4.3. การเซ็นร่วมกัน ซึ่งลำดับการเซ็นไม่สำคัญ ตรวจสอบได้โดยดูที่เอกสารชุด (version)ว่ามีใครเซ็นไปบ้างแล้ว และยังเหลือใครอยู่ (ดูจาก AUTHOR และ STATUS = SIGNED)

4.4. การเซ็นลำดับชั้น ก่อนการเซ็นต้องมีการเช็คว่าผู้ที่มีลำดับน้อยกว่าเซ็นหรือยัง ถ้ายังจะไม่สามารถเซ็นได้ โดยดูลำดับการเซ็นได้จากต้นไม้กำหนดสิทธิ์ (TREE OF AUTHORIZE) ในเอกสารกำหนดสิทธิ์ภายใต้ <SIGNER> TAG แต่ถ้าคนมีสิทธิ์เซ็นไม่อยู่ สามารถมอบอำนาจให้คนอื่นแทนได้ในช่วงเวลาที่กำหนด



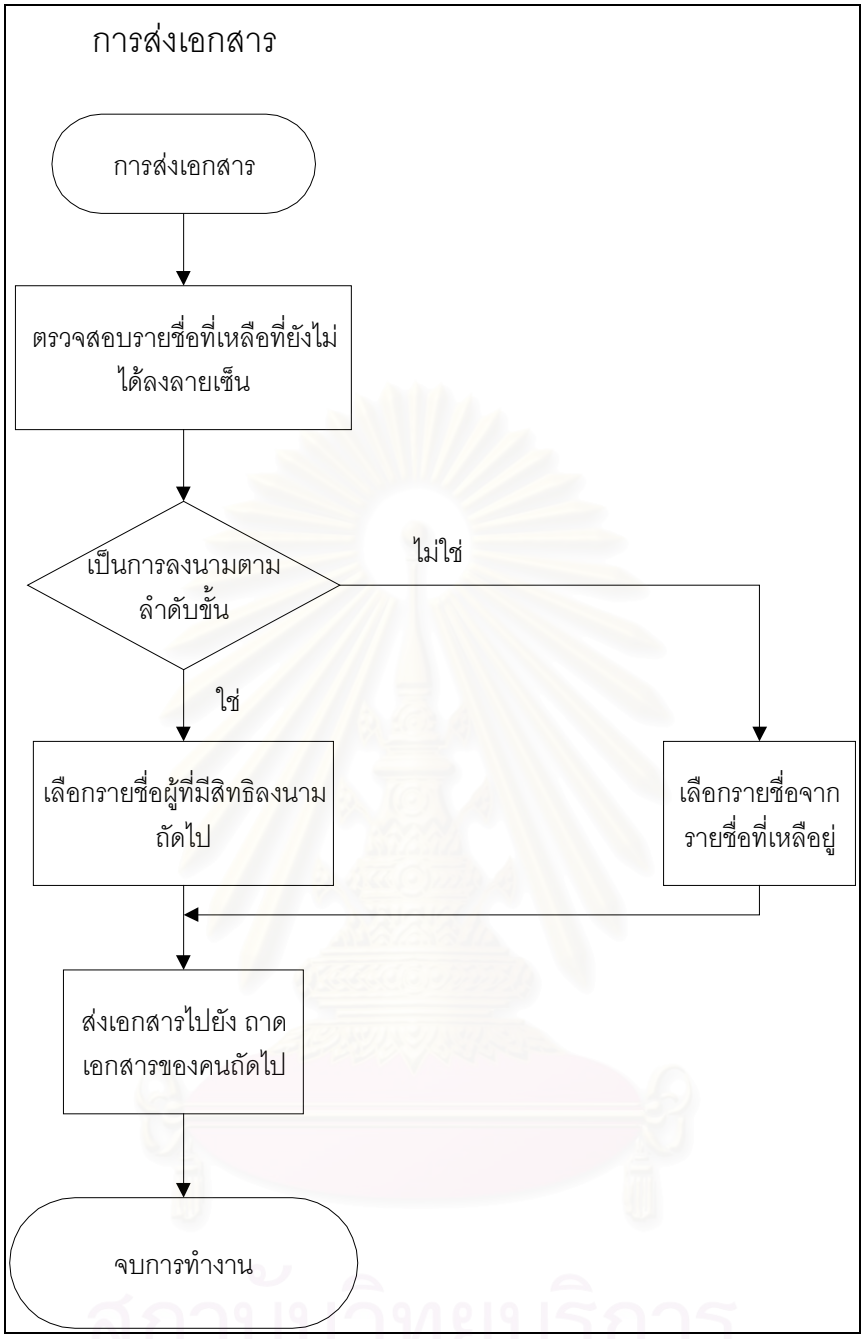
รูปที่ 5.3 ขั้นตอนการลงนามเอกสาร

5. การแนบเอกสารคือ ความต้องการแนบเอกสารอื่นเพื่อประกอบการพิจารณาเช่น แนบใบเสนอราคาหรือใบอนุมัติการแต่งตั้ง



รูปที่ 5.4: ขั้นตอนการแนบเอกสาร

6. การส่ง/รับเอกสาร เป็นฟังก์ชั่นที่ใช้ในการส่งเอกสารที่ลงนามแล้วไปยังบุคคลอื่นที่เกี่ยวข้อง โดยตรวจดูรายชื่อได้จากรายชื่อผู้มีสิทธิ์ลงนามในเอกสารและยังไม่ได้ลงนาม ส่วนการรับเอกสารเป็นการเรียกดูเอกสารที่ส่งมาถึงตัวเอง เพื่อทำการลงนามหรือแก้ไขเอกสาร



รูปที่ 5.5: ขั้นตอนการส่งเอกสาร

รูปที่ 5.6: ขั้นตอนการรับเอกสาร

วงจรชีวิตของเอกสารที่อธิบายในบทนี้จะครอบคลุมเฉพาะในส่วนที่เกี่ยวข้องกับตัวเอกสารเท่านั้น ในส่วนของพฤติกรรมของระบบในแง่ผู้ดูแลระบบจะกล่าวถึงในบทต่อไป

บทที่ 6

ขั้นตอนของระบบในแง่ของผู้ดูแลระบบ

ในบทนี้จะอธิบายถึงพฤติกรรมของระบบในแง่ของผู้ดูแลและพัฒนาระบบเอกสารที่มีการลงนามซึ่งต่างกับบทที่ 5 ที่อธิบายเฉพาะวงจรชีวิตของเอกสารที่มีการลงนามในแง่ของผู้ใช้งานเอกสาร โดยสามารถแบ่งหน้าที่หลักได้ดังต่อไปนี้

1. การสร้างเอกสาร
2. การป้องกันการแก้ไข
3. การเพิ่ม / ลดสิทธิ์ผู้ใช้งานเอกสารแต่ละเอกสาร
4. การยกเลิกเอกสาร
5. การทำลายเอกสาร
6. การทำสำเนา
7. การจัดการกับความผิดพลาดของเอกสาร
8. การตรวจสอบการแก้ไขเอกสาร
9. การกำหนดสิทธิ์ของผู้ใช้ระบบ
10. การเข้าสู่ระบบ

โดยมีรายละเอียดดังนี้

1. การสร้างเอกสารใหม่ คือการสร้างชุดเอกสารที่มีการลงนาม โดยใน 1 ชุดของเอกสารประกอบด้วยเอกสารหลักและเอกสารรองอีก 3 เอกสารดังที่ได้กล่าวไว้แล้ว โดยการสร้างตัวเอกสารมีขั้นตอนดังต่อไปนี้

1.1. สร้างเอกสารหลักเอกซ์เอ็มแอล (XML) นำเอกสารต้นฉบับที่เป็นเอกสารหลักมาแบ่งเป็นส่วนต่างๆ ของเอกสารตามลักษณะใช้งานจริง (เช่น ตาราง แถว ส่วนการลงนาม) โดยใช้แทก (TAG) เป็นตัวกำหนดส่วนของเอกสาร และใส่ตัวชี้ (Pointer) ในแต่ละส่วนของเอกสารหลักไปยังเอกสารประกอบทั้ง 3

1.2. สร้างเอกสารกำหนดสิทธิ์ เพื่อสร้างกลุ่มของผู้มีสิทธิ์ใช้เอกสารที่สร้างใน 1.1 และกำหนดสิทธิ์การใช้ให้ผู้ใช้แต่ละคน

1.3. สร้างเอกสารกำหนดการลงนาม สร้างชื่อของผู้มีสิทธิ์ลงนามในแต่ละส่วนที่อ้างถึงจากเอกสารหลัก (1.1) รวมถึงกำหนด คุณสมบัติต่างๆ เช่น กระบวนการทำงาน (Algorithm) กระบวนการหาค่าแฮช (Hash Algorithm)

1.4. สร้างเอกสารกำหนดชุด (version) ใส่ชื่อของผู้สร้างใน <VERSION_TAG> เพื่อกำหนดเป็นชุด (version)ที่ 1.0 กำหนดวันที่สร้าง และชื่อผู้สร้าง

หมายเหตุ ในวิทยานิพนธ์ฉบับนี้สร้างเอกสารครั้งแรกด้วยมือ โดยใช้โปรแกรมสร้างเอกสารทั่วไป (Text Editor) ในการสร้าง แทก ต่างๆ

2. การป้องกันการแก้ไขเอกสาร เนื่องจากเอกสารหลักและเอกสารเสริมทั้ง 3 เอกสารเป็นข้อความตัวอักษรทั้งหมด ไม่ได้เข้ารหัสเป็นรูปแบบพิเศษ ดังนั้น จึงต้องมีการกำหนดความปลอดภัยในการเข้าถึงแฟ้มข้อมูล โดยมีการจัดผู้ที่มีสิทธิ์ใช้แฟ้มข้อมูลในส่วนนั้น โดยใช้ความสามารถของระบบปฏิบัติการเป็นตัวจัดการในการกำหนดสิทธิ์ในการเข้าถึงข้อมูล โดยการเรียกฟังก์ชั่นการเข้าถึงแฟ้มข้อมูลของระบบปฏิบัติการ เช่น เอ็นทีเอฟเอส (NTFS) หรือ ยูนิกซ์ (UNIX) ในการกำหนดสิทธิ์ ระบบจึงต้องมีการตรวจสอบว่าผู้ใช้ระบบคนใดมีสิทธิ์ในการใช้แฟ้มข้อมูลใด แต่โดยปกติผู้มีสิทธิ์แก้ไข คือ ผู้ดูแลระบบ (System Admin) เท่านั้น เพื่อป้องกันการมาแก้ไขเอกสารในส่วนที่ตัวเองไม่ได้เป็นเจ้าของ

3. การเพิ่ม / ลด สิทธิ์ผู้ใช้เอกสารในแต่ละเอกสาร ในแง่สิทธิ์ของผู้ใช้นั้นสามารถแยกได้เป็น 2 ระดับ คือ

3.1. ระดับของผู้ดูแลระบบ (System Admin) หมายถึงว่าใครมีสิทธิ์ในการแก้ไขเอกสารทั้งหมด โดยใช้โปรแกรมสร้างเอกสาร (Text Editor) หรือการทำปฏิบัติการ (Operation) ต่างๆ เช่น การทำสำเนาข้อมูล การตรวจสอบการแก้ไขเอกสาร ซึ่งเป็นการกระทำกับระบบเอกสารที่มีการลงนามไม่ใช่ที่ตัวโครงสร้างของเอกสาร ในระดับนี้ถูกกำหนดโดยผู้พัฒนาระบบแล้วสร้างและเก็บอยู่ในตัวระบบ

3.2. ระดับของผู้ใช้เอกสารที่มีการลงนาม (Document User) คือผู้ที่มีสิทธิ์ในการแก้ไขตัวเอกสารโดยชื่อจะเก็บอยู่ในเอกสารกำหนดสิทธิ์เมื่อมีการให้ / ลดสิทธิ์ในการลงนาม แก้ไขเอกสาร ระบบเอกสารจะทำหน้าที่เพิ่มชื่อผู้มีสิทธิ์ในเอกสารกำหนดสิทธิ์ผู้ใช้ นำผู้ใช้มาเป็นสมาชิกของกลุ่มที่มีสิทธิ์เหมือนกัน (ถ้ามี) <GRP> กำหนดต้นไม้ของการกำหนดสิทธิ์ (TREE OF AUTORIZE) โดยจัดอยู่ในกลุ่มที่ต้องมีการเซ็นลงนามตามลำดับขั้น หรือจัดตามโครงสร้างขององค์กรนั้นๆ ส่วนการลดสิทธิ์ผู้ใช้สามารถทำในลักษณะเดียวกับการเพิ่มผู้ใช้

4. การยกเลิกเอกสาร เอกสารที่ยกเลิกไม่ได้ถูกลบออกจากระบบเพียงแต่ถูกตั้งสถานะเป็นยกเลิก (CANCELLED) ในเอกสารกำหนดชุด (version)เอกสารที่ถูกยกเลิกไม่สามารถแก้ไขได้อีกต่อไปแต่สามารถเรียกดูเอกสารได้ในแต่ละชุด (version)ว่ามีการแก้ไขอย่างไรบ้าง

5. การลบและการทำลายเอกสาร การลบเอกสารต่างจากการยกเลิกเอกสาร คือ เอกสารที่ถูกลบจะไม่สามารถเรียกดูได้ต่อไป จะดูได้แต่ในส่วนการตรวจสอบเอกสาร ว่าใครเป็นผู้ลบเอกสาร แต่ตัวเอกสารยังไม่ถูกลบออกจากระบบ เพียงแต่ย้ายไปเก็บไว้ที่อื่นและสามารถเรียกกลับคืนมาได้ โดยผู้ที่มีสิทธิ์เท่านั้น ส่วนการทำลายเอกสาร คือ การลบแฟ้มข้อมูลทั้งหมดทิ้งเหลือเพียงแต่เอกสารที่ใช้ในการตรวจสอบเท่านั้น เพื่อใช้ในการติดตามดูว่าใครทำอะไรกับเอกสารไปบ้างก่อนการทำลาย เอกสารที่ถูกทำลายจะไม่สามารถเรียกกลับมาได้อีก

6. การทำสำเนาเอกสาร คือ การสร้างเอกสารอีกชุดที่เหมือนกัน แต่เอกสารที่เป็นสำเนาจะไม่สามารถแก้ไขได้อีก และมีเอกสารตรวจสอบของเอกสารสำเนา เนื่องจากเอกสารสำเนามีวัตถุประสงค์เพื่อส่งให้คนอื่นไว้อ้างอิงหรือดูเท่านั้น

7. การจัดการกับความผิดพลาดของเอกสารและความผิดพลาดในระบบ (EXCEPTION HANDLER) ระบบต้องมีความสามารถในการตรวจสอบความผิดพลาดในเอกสารได้ เช่น ผู้ใช้มีสิทธิ์ลงนามในเอกสารกำหนดการลงนาม แต่ไม่มีชื่อในเอกสารกำหนดสิทธิ์ การอ้างอิงระหว่างเอกสารหลัก ไม่สามารถอ้างอิงได้ ถ้า

เกิดความผิดพลาดในเอกสารผู้ที่มีอำนาจแก้ไขเอกสารได้มีเพียงคนเดียว คือ ผู้สร้างเอกสาร แม้แต่ผู้ดูแลระบบไม่สามารถแก้ไขได้ ทั้งนี้เพื่อป้องกันไม่ให้คนอื่นมาแก้ไขเอกสาร

ส่วนความผิดพลาดอันเกิดจากระบบนั้น อย่างเช่น หาเอกสารไม่เจอ ไม่สามารถเข้ารหัสข้อมูลได้ เนื่องจาก กุญแจที่ใช้ไม่ถูกต้อง ระบบจะหยุดทำงานทันที และผู้ที่สามารถสั่งแก้ไขได้จะมีเพียง 2 คน คือ ผู้ดูแลระบบ (System Admin) และผู้เป็นเจ้าของเอกสาร

8. การตรวจสอบการแก้ไขของเอกสาร ในการตรวจสอบการแก้ไขของเอกสารแบ่งได้เป็น 2 ส่วนคือ

8.1. การตรวจสอบที่ระดับของเอกสาร ว่ามีใครมาทำอะไรกับเอกสารบ้าง เช่น การเปลี่ยนข้อความ การลงนาม การแก้ไขที่ระดับเอกสารสามารถตรวจสอบได้ที่เอกสารกำหนดชุด (version) (VERSION_TAG)

8.2. การตรวจสอบที่ระบบของเอกสาร ในส่วนนี้มีการเก็บการกระทำกับเอกสารที่ไม่ได้เก็บในข้อ 8.1 ไว้ในแฟ้มข้อมูล AUDIT LOG โดยมีโครงสร้างหลักของการเก็บดังนี้

OPERATION	NAME	DATE/TIME	REMARKS

OPERATION	คือ คำสั่งหรือการกระทำที่ทำกับเอกสาร เช่น การเข้ารหัสข้อมูล การสร้างเอกสาร การลบเอกสาร การเข้าสู่ระบบ
NAME	คือ ชื่อผู้ที่ทำปฏิบัติการ (OPERATION)
DATE/TIME	คือ วันที่และเวลาที่ทำปฏิบัติการ (OPERATION)
REMARKS	คือ รายละเอียดของการกระทำ (ถ้ามี)

9. การกำหนดสิทธิ์ของผู้ใช้งาน คือ การกำหนดว่าใครมีสิทธิ์ในการทำอะไรในระบบ (ฟังก์ชั่นการทำงานทั้งหมดของระบบ) ซึ่งเป็นคนละส่วนกับสิทธิ์ในการใช้เอกสารที่กำหนดในเอกสาร แต่ว่ามีผู้มีสิทธิ์แก้ไขเอกสารต้องมีสิทธิ์ในการใช้ระบบด้วย แต่ผู้มีสิทธิ์ในการใช้ระบบอาจไม่มีสิทธิ์ในการแก้ไขเอกสาร ผู้ใช้ระบบสามารถจัดกลุ่มผู้มีสิทธิ์คล้ายกันเป็นกลุ่มเดียวกัน เช่น กลุ่มของผู้ดูแลระบบมีสิทธิ์เต็มที่ในการกระทำกับระบบ กลุ่มผู้ใช้มีสิทธิ์ในการใช้ระบบเท่านั้น แต่ไม่สามารถลบ / ทำลายเอกสารได้ เป็นต้น ในส่วนนี้รวมถึงการสร้าง ลบผู้ใช้ระบบ การเก็บ / เปลี่ยนรหัสผ่านในการเข้าระบบ

10. การเข้าสู่ระบบ (logon) เป็นฟังก์ชั่นแรกของระบบที่ผู้ใช้ต้อง logon ก่อนเข้าสู่ระบบได้ โดยในการ logon ระบบจะตรวจสอบชื่อผู้ใช้ (login ID) กับฐานข้อมูลว่ามีจริงหรือไม่ และรหัสผ่านถูกต้องหรือไม่ ถ้าถูกต้องระบบจะทำการดูว่าผู้ใช้คนนี้มีสิทธิ์อะไรบ้าง จะทำการมอบหมายสิทธิ์ ให้แก่ผู้ใช้โดยสิทธิ์นี้เป็นของตอน logon ขณะนั้น ถ้าผู้ดูแลระบบเปลี่ยนแปลงสิทธิ์ในขณะที่ผู้ใช้อยู่ในระบบอยู่ จะไม่มีผลจนกว่าผู้ใช้จะเข้าสู่ระบบใหม่อีกครั้ง ทั้งนี้เพื่อลดภาระในการตรวจสอบสิทธิ์ทุกครั้งที่มีการเรียกใช้ฟังก์ชั่นของระบบ

ในส่วนของหน้าที่ที่กล่าวมาจะเป็นหน้าที่ของระบบที่ทำงานอยู่บนตัวเอกสารอีกที เพื่อทำให้ระบบสามารถนำไปใช้งานได้จริง เช่น การสร้าง ลบ ทำสำเนาเอกสาร จากบทนี้จะนำไปสู่การออกแบบระบบต้นแบบเพื่อใช้ในการทดสอบโครงสร้างเอกสารที่ออกแบบว่าสามารถรองรับความต้องการได้ดีเพียงใด

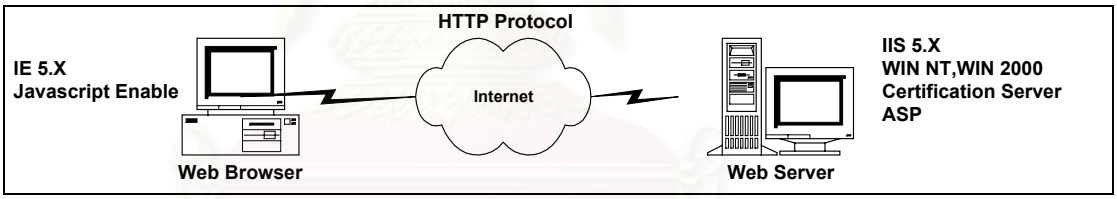
บทที่ 7

การออกแบบระบบต้นแบบ

ระบบต้นแบบของการจัดการเอกสารที่มีการลงนามเป็นระบบที่ใช้ในการรับส่งเอกสารที่มีการลงนามตลอดจนการกระทำต่างๆ กับเอกสาร เช่น การแก้ไขเอกสาร การเรียกดูเอกสารในแต่ละชุด (version) และการลงนามเอกสาร จุดประสงค์ของการสร้างระบบต้นแบบทำให้สามารถทดสอบโครงสร้างของเอกสารที่ออกแบบว่าสามารถใช้งานได้ดีเพียงใดในการนำมาใช้งานกับโปรแกรมประยุกต์ต่างๆ ทั้งนี้เพื่อให้สามารถนำโครงสร้างของเอกสารที่ออกแบบไว้ไปใช้ในลักษณะงานที่แตกต่างกัน เช่น ระบบงานการจัดซื้อสินค้า ระบบงานบุคคล รวมถึงการอนุมัติต่างๆ

7.1 โครงสร้างรวมของระบบ

ในการออกแบบระบบต้นแบบเลือกใช้เทคโนโลยีเว็บเพจในการสร้างระบบ เนื่องจากผู้ใช้สามารถใช้จากที่ใดก็ได้ โดยเฉพาะในส่วนของผู้ใช้ (client) ไม่จำเป็นต้องมีโปรแกรมเฉพาะเพื่อใช้ในการทำงาน ทำให้สะดวกในการใช้งาน โครงสร้างของระบบประกอบด้วยส่วนหลักๆ ดังรูป คือ



รูปที่ 7.1: โครงสร้างของระบบ

เทคโนโลยีที่ใช้ในการพัฒนาคือ เอเอสพี (ASP) ทั้งนี้เพราะความสะดวกเนื่องจากเอเอสพี เขียนได้ง่ายและมากับไอไอเอส (IIS) ส่วนทางด้านผู้ใช้ใช้จาวาสคริปต์ (Java Script) ในการทำส่วนของด้านผู้ใช้ (Client)

เนื่องจากการออกแบบระบบต้นแบบเป็นลักษณะของสถาปัตยกรรมหลายชั้น (N-tier architecture) โดยสามารถแบ่งการทำงานออกเป็น 3 ส่วนหลักๆ คือ

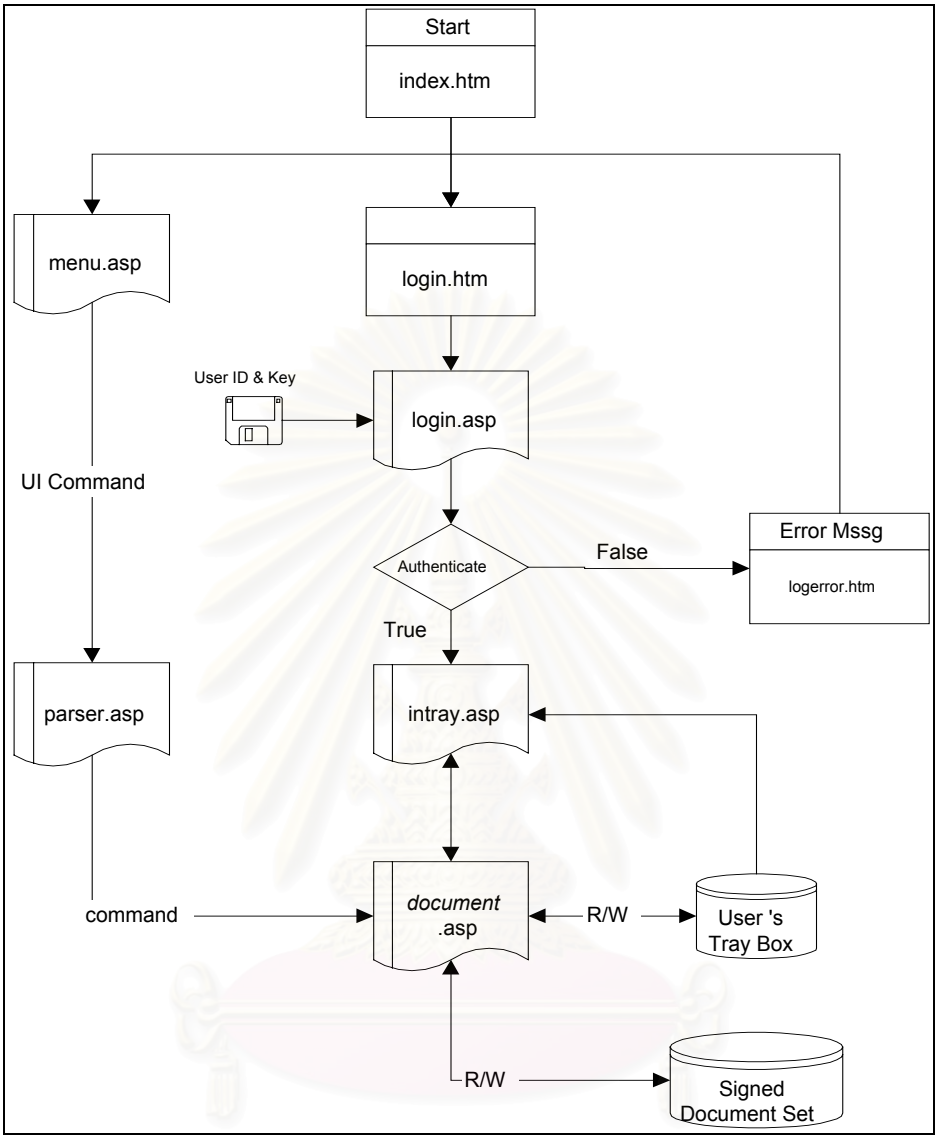
- ส่วนแสดงผล (Presentation) ใช้ในส่วนของการแสดงเอกสารเอกซ์เอ็มแอล (XML) การรับคำสั่งจากเมนูซึ่งอยู่ในฝั่งของผู้ใช้ ในที่นี้ใช้โปรแกรมอินเทอร์เน็ตเอกซ์พลอเลอร์ ชุด (version) 5
- ส่วนตรรกศาสตร์ของโปรแกรม (Business Logic) คือส่วนที่ทำหน้าที่กำหนดการทำงานร่วมกันของโปรแกรม เช่น การส่งเอกสารไปให้ใคร ตรวจสอบถาดเอกสารของแต่ละคน การติดต่อกับแฟ้มข้อมูลเอกซ์เอ็มแอล (XML) และโครงสร้างแฟ้มข้อมูลของถาดเอกสาร

- ส่วนเก็บข้อมูล (File Server) คือส่วนที่ทำหน้าที่เก็บชุดของเอกสารที่มีการลงนาม และถาดเอกสารของแต่ละคน ส่วนนี้เก็บอยู่บนเครื่องบริการ (Server) เพื่อให้ส่วนอื่นสามารถเรียกใช้และแก้ไข

ตารางที่ 7.1: แสดงการทำงานส่วนต่างๆ ของระบบ

PRESENTATION	BUSINESS LOGIC	DATA FILE SERVER
• รับคำสั่งจากเมนู • การแสดงผลสถานะต่างๆ ในเมนู • การเข้าระบบ • การแสดงเอกสาร XML o ตามชุด (version) o ตามสิทธิ์ของแต่ละคน • ส่งเอกสารไปยัง Server	• ตรวจสอบผู้ใช้ระบบ Authenticate • การลงนามเอกสาร • การจัดการถาดเอกสาร (In-tray) • การแก้ไขเอกสาร XML • การเข้ารหัส / ถอดรหัสข้อมูล • การควบคุมเอกสารไม่ให้มีคนใช้งานมากกว่า 1 คนในเวลาเดียวกัน	• ชุดเอกสาร XML o MAIN.XML o SECURE.XML o SIGNATURE.XML o VERSION.XML o MAIN.ASP o DOCUMENT.HTM *DOCUMENT* หมายถึงชื่อเอกสารที่มีการลงนาม เช่น Payslip • ถาดเอกสารของแต่ละคนอยู่ใน *Web_Directory*/TRAY/*Username*

สามารถแสดงผังงานหลักของโปรแกรมได้ดังนี้



รูปที่ 7.2: ผังงานของโปรแกรม

โดยโปรแกรมเริ่มจากการตรวจสอบผู้ใช้ (login.asp) โดยการเปรียบเทียบรหัสผ่านกับรหัสที่เก็บอยู่ในแผ่นรหัส (อาจเป็น Floppy Disk CD ROM) ถ้าถูกต้องจะให้ใช้งานระบบ จากนั้นโปรแกรมจะเปิดถาดเอกสารของผู้ใช้ (INTRAY.ASP) เพื่อดูว่ามีเอกสารอะไรที่เข้ามาใหม่บ้าง เมื่อคลิกที่เอกสารที่ต้องการเปิดจะมีโปรแกรมเอเอสพี (ASP) ที่ชื่อเดียวกับเอกสารอยู่เสมอ โปรแกรมนี้มีการติดต่อกับแฟ้มข้อมูลต่างๆ บนเซิฟเวอร์ (Server)

เมื่อผู้ใช้ส่งคำสั่งผ่านเมนู (MENU.ASP) คำสั่งถูกส่งไปยังโปรแกรม Dispatch (dispatch.asp) เพื่อทำการแปลงคำสั่งและกำหนดค่าต่างๆ ก่อนส่งให้ *document*.asp จัดการ *document*.asp (document คือแบบฟอร์มที่เปิดใช้งาน) จะรับคำสั่งจาก parser.asp และกระทำตามคำสั่ง ซึ่งมีการทำงานเกิดขึ้นทั้งในส่วนของ client (ใช้ Java Script) และส่วนของ Server (ใช้ ASP) ขึ้นกับฟังก์ชั่นการทำงาน เป็นอย่างนี้เรื่อยไปตาม

วงจรชีวิตของเอกสารที่กล่าวมาแล้วจนกว่าผู้ใช้จะออกจากระบบหรือส่งเอกสารที่แก้ไข ลงนามแล้ว ไปให้บุคคลอื่นต่อไป

7.2 ส่วนติดต่อกับผู้ใช้

ในส่วนเมนูที่ติดต่อกับผู้ใช้สามารถแบ่งได้เป็น 3 ส่วนหลักๆ คือ

- **เมนูคำสั่งในลักษณะดึงลงมา** (Pull Down)
- ส่วนคำสั่งที่เป็นลักษณะของแถบเครื่องมือ (Tool Bar) เป็นไอคอน (icon)
- การแสดงผลสถานะของเอกสารเพื่อแสดงสถานะและคุณลักษณะของเอกสาร

รูปที่ 7.1: แสดงส่วนติดต่อกับผู้ใช้

สามารถอธิบายส่วนต่างๆ ได้ดังนี้

- เมนู Document ทำหน้าที่จัดการเกี่ยวกับเอกสารมีเมนูย่อยๆ ดังต่อไปนี้
  - ATTACH ใช้ในการแนบเอกสารอื่นพร้อมกับเอกสารฉบับที่เปิดอยู่ โดยผู้ใช้ต้องใส่ชื่อแฟ้มข้อมูลและที่อยู่ของแฟ้มข้อมูลที่ต้องการแนบ
  - CREATE คือการสร้างเอกสารขึ้นมาใหม่ (ยังไม่พัฒนาในวิทยานิพนธ์ฉบับนี้ แต่จะเชื่อมกับระบบที่ใช้สร้างเอกสารลงนามในภายหลัง)
  - OPEN คือการเปิดเอกสาร โดยระบุชื่อแฟ้มข้อมูล สารบบ (Directory)ที่ต้องการเปิด
  - SAVE เก็บเอกสารที่แก้ไข / เปิดอยู่ลงในแฟ้มข้อมูล
  - SEND ส่งเอกสารไปให้ผู้อื่น โดยระบบสามารถกำหนดชื่อผู้ที่ควรเป็นผู้รับรายต่อไป (ในกรณีของการเซ็นแบบลำดับขั้น)
  - INBOX คือการเปิดถาดเอกสารของผู้ที่ใช้ระบบ
- เมนู EDIT คือเมนูที่ใช้ในการแก้ไขดัดแปลงเอกสาร ประกอบด้วยเมนูย่อยๆ ดังต่อไปนี้
  - COPY คือ การทำสำเนาเอกสารอีกชุด
  - CANCEL คือ การยกเลิกเอกสารฉบับที่เปิดอยู่
  - DELETE คือ การลบเอกสารและไม่สามารถเรียกคืนได้
  - RECOVER คือ การเรียกคืนเอกสาร ใช้ได้เฉพาะเอกสารที่โดน Purge อยู่
  - PURGE คือ การลบเอกสารแต่จะสามารถเรียกคืนได้โดยคำสั่ง RECOVER

- UPDATE คือ การแก้ไขเปลี่ยนแปลงเอกสาร ใช้เมนูนี้หลังจากที่มีการแก้ไขเอกสารเรียบร้อยแล้ว
- SUBMIT คือ ยืนยันการแก้ไขเอกสาร

- เมนู VIEW คือ เมนูที่ใช้ดูสถานะของเอกสารหรือค่าต่างๆ ที่อยู่ในเอกสาร
  - Document ดูเอกสารทั้งฉบับ
  - Signed By ดูว่ามีใครเซ็นชื่อกำกับเอกสารบ้างและเมื่อใด
  - History ดูว่ามีการแก้ไขอะไร โดยใครบ้าง
  - Right ดูผู้มีสิทธิ์ในการแก้ไขเอกสาร ดูว่าใครมีลำดับในการลงนาม
- เมนู Signature คือเมนูที่เกี่ยวข้องกับการลงลายเซ็น
  - SIGNED คือการลงนามรับรองเอกสาร
  - INSERT KEY คือการใส่ Public Key ลงในเอกสารเพื่อใช้ในการตรวจสอบเอกสาร
- เมนู Admin คือฟังก์ชั่นที่เกี่ยวกับการดูแลระบบ
  - AUDIT READ คือการอ่านแฟ้มข้อมูลการตรวจสอบ (Audit) เอกสาร
  - AUDIT WRITE คือการกำหนดให้เขียนแฟ้มข้อมูล การตรวจสอบของเอกสารลงในแฟ้มข้อมูลที่กำหนด
  - USER คือการกำหนดสิทธิ์ของผู้ใช้ระบบ

ในส่วนของแถบเครื่องมือได้กำหนดไว้สำหรับคำสั่งที่ใช้งานบ่อย ๆ ได้แก่


ไปที่ log on ของระบบ ระบบจะ logoff โดยอัตโนมัติ


คือ การดูเอกสาร เทียบเท่ากับเมนู View → Document


คือ การลงนามรับรองเอกสาร


คือ ยืนยันการแก้ไขเอกสาร


คือ การส่งเอกสาร


คือ การแนบ (Attach) เอกสาร


คือ การตั้งค่าของระบบ (option)


คือ การดูถาดเอกสาร

คือ การเลือกชุด (version)ในการแสดงผลของเอกสาร

ส่วนแสดงผลสามารถแสดงได้ดังนี้

แสดงสิทธิ์การใช้งานเอกสารของผู้ใช้ระบบ

R คือการอ่านเอกสาร

W คือการเขียนเอกสาร (การเขียนใหม่)

S คือการลงนามเอกสาร

A คือการแนบเอกสาร

M คือการแก้ไขเอกสาร

แสดงชื่อผู้แก้ไขเอกสาร

7.3 **ฟังก์ชั่นของระบบ**

ในการออกแบบระบบยึดหลักในส่วนของการแยกฟังก์ชั่นออกเป็นส่วนย่อยๆ เพื่อสามารถนำฟังก์ชั่นไปใช้กับโปรแกรมอื่นๆ ได้ โดยระบบอื่นสามารถนำโครงสร้างเอกสารและฟังก์ชั่น ของระบบที่ใช้กับเอกสารไปประยุกต์ใช้งานในด้านต่างๆ เช่น ระบบบัญชีที่ต้องมีการอนุมัติโดยใช้ลายเซ็น ระบบใบสั่งซื้อเป็นต้น

โดยฟังก์ชั่นของระบบสามารถแบ่งได้เป็น 2 ส่วนหลักๆ คือ

1. ฟังก์ชั่นที่ใช้งานกับเอกสารที่มีการลงนาม คือ ฟังก์ชั่นที่ออกแบบมาเพื่อใช้กับโครงสร้าง เอกสารลงนาม สามารถนำฟังก์ชั่นเหล่านี้ไปใช้กับระบบอื่นๆ ได้
2. ฟังก์ชั่นที่ใช้กับระบบทดสอบเอกสารที่มีการลงนามเป็นฟังก์ชั่นที่ออกแบบมาเพื่อใช้กับระบบทดสอบเท่านั้น ไม่สามารถนำไปใช้งานอื่นๆ ได้

7.3.1 **ฟังก์ชั่นที่ใช้งานกับเอกสารที่มีการลงนาม**

สามารถแบ่งย่อยเป็นกลุ่มหลักๆ ได้ดังนี้

- ฟังก์ชั่นการอ่านค่าเอกสาร
- ฟังก์ชั่นการแก้ไขเอกสาร
- ฟังก์ชั่นการตั้งค่าคุณสมบัติ (Attribute) ของเอกสาร
- ฟังก์ชั่นการอ่านค่าคุณสมบัติ (Attribute) ของเอกสาร

7.3.1.1 ฟังก์ชั่นการอ่านค่าเอกสาร

คือ การอ่านค่าของเอกสารเอกซ์เอ็มแอล (XML) โดยการระบุแทก (TAG) ที่ต้องการอ่าน โดยมีโครงสร้างดังต่อไปนี้

VALUE = **READ_DOC** (*VERSION*, *TAG*, *DOC_NAME*)

TAG (String) = TAG ของเอกสาร XML ที่ต้องการหา

VERSION (Number) = คือชุด (version)ของเอกสารที่ต้องการหา

VALUE (String) = ข้อความเอกสารที่อ่านได้

DOC_NAME (String) = ชื่อชุดเอกสารที่ต้องการอ่าน

โดย READ_DOC ทำการอ่านค่าของแทกเอกซ์เอ็มแอล (XML) ที่ต้องการ โดยการกำหนดแทกที่ต้องการ ชุด (version)ที่ต้องการอ่านจากชุดเอกสารที่ต้องการอ่าน

7.3.1.2 ฟังก์ชั่นการแก้ไขเอกสาร

คือการแก้ไข / เขียน ข้อมูลลงในแทกที่ต้องการโดยมีการกำหนดแทก ผู้ที่เขียนและชื่อชุดเอกสารที่ต้องการเขียน

STATUS = **WRITE_DOC** *(USER, TAG, TEXT, DOC_NAME)*

STATUS (Number) = สถานะการทำงานของฟังก์ชั่น

TAG (String) = ชื่อแทกที่ต้องการเขียน

TEXT(String) = ชื่อข้อความที่ต้องการเขียน

DOC_NAME (String) = ชื่อชุดของเอกสาร

USER (String) = ชื่อผู้ทำการแก้ไข

7.3.1.3 ฟังก์ชั่นการตั้งค่าคุณสมบัติของเอกสาร

คือฟังก์ชั่นที่ใช้ในการแก้ไขคุณสมบัติของเอกสาร เช่น การแก้ไขชุด (version)ของเอกสาร การลงนามเอกสาร

STATUS = **ATTACH** *(ATTACH_FILE, DOC_NAME)*

STATUS (Number) = สถานะการทำงานของฟังก์ชั่น

ATTACH_FILE = ชื่อแฟ้มข้อมูลที่ต้องการแนบรูปแบบ url

DOC_NAME (String) = ชื่อชุดของเอกสาร

ฟังก์ชั่นนี้ใช้ในการแนบเอกสาร

STATUS = **SIGNED** *(USER, TAG, DOC_NAME)*

STATUS (Number) = สถานะการทำงานของฟังก์ชั่น

TAG (String) = ชื่อแทกที่ต้องการเขียน

DOC_NAME (String) = ชื่อชุดของเอกสาร

USER (String) = ชื่อผู้ทำการแก้ไข

ฟังก์ชั่นนี้ใช้ในการลงนามเอกสาร

STATUS = **INSERT_KEY** *(USER, KEY, DOC_NAME)*

STATUS (Number) = สถานะการทำงานของฟังก์ชั่น

KEY (String) = กุญแจรหัส

DOC_NAME (String) = ชื่อชุดของเอกสาร

USER (String) = ชื่อผู้ทำการแก้ไข

ฟังก์ชั่นนี้ใช้ในการใส่กุญแจรหัสสาธารณะ

7.3.1.4 ฟังก์ชั่นการอ่านค่าคุณสมบัติของเอกสาร

คือฟังก์ชั่นที่ใช้ในการอ่านค่าคุณสมบัติของเอกสาร เช่น สิทธิ์ในการใช้งานเอกสาร จำนวนผู้ลงนาม

RIGHTS = **READ_PERMISSIONS** (*USER*, *TAG*, *DOC_NAME*)

RIGHTS (Number) สิทธิ์ของเอกสาร โดยมีค่าดังนี้

READ	=	1
WRITE	=	2
MODIFY	=	4
SIGNATURE	=	8
ATTACH	=	16
COPY	=	32

สามารถหาสิทธิ์ของเอกสารได้โดยการนำค่า RIGHTS ที่ได้มาทำการ OR กันเช่น มีสิทธิ์ READ และ SIGNATURE = 1 + 8 = 9

DOC_NAME (String) = ชื่อชุดของเอกสาร

TAG (String) = ชื่อแทกที่ต้องการทราบสิทธิ์

ฟังก์ชั่นนี้ใช้ในการอ่านสิทธิ์ของแทกที่ต้องการ

USERLIST = **READ_SIGNED** (FLAG, *TAG*, *DOC_NAME*)

USERLIST (Array of String)= รายชื่อผู้ใช้ระบบ

FLAG (String) = กำหนดการอ่านค่าการลงนาม

SIGNED	ผู้ที่ลงนามแล้ว
ALL	รายชื่อทั้งหมดที่มีสิทธิ์ลงนาม
NO_SIGNED	ผู้ที่ยังไม่ได้ลงนาม

DOC_NAME (String) = ชื่อชุดของเอกสาร

TAG (String) = ชื่อแทกที่ต้องการทราบการลงนาม

หมายเหตุ TAG ในที่นี้อาจเป็นได้ทั้งเอกสารหรือส่วนของเอกสาร

7.3.2 ฟังก์ชั่นที่ใช้กับระบบทดสอบเอกสาร

สามารถสรุปฟังก์ชั่นที่ใช้กับระบบทดสอบเอกสาร ที่ทดลองสร้างขึ้นได้ดังนี้ โดยแบ่งตามกลุ่มฟังก์ชั่นการทำงานดังตารางที่ 7.2 ดังนี้

ตารางที่ 7.1: สรุปฟังก์ชั่นการทำงานของระบบเอกสารที่มีการลงนาม

กลุ่มฟังก์ชั่นการทำงานของระบบ	ชื่อฟังก์ชั่น	รายละเอียด
1. การสร้างเอกสาร	CREATE_SIGNED_DOC - CREATE_MAIN - CREATE_SECURITY - CREATE_SIGNATURE - CREATE_VERSION OPEN_DOCUMENT SAVE_DOCUMENT	สร้างเอกสารใหม่ สร้างเอกสารหลัก สร้างเอกสารกำหนดสิทธิ์ สร้างเอกสารกำหนดการลงนาม สร้างเอกสารกำหนดชุด (version) เปิดเอกสาร บันทึกเอกสาร
2. การป้องกันการแก้ไขเอกสาร	DOCUMENT_PERMISSION	เช็คสิทธิ์การเข้าถึงแฟ้มข้อมูล (ระดับ OS)
3. การเพิ่ม / ลด สิทธิ์ผู้ใช้ระบบ	MODIFY _RIGHT	แก้ไข / สร้างสิทธิ์ของผู้ใช้/ กลุ่มผู้ใช้
4. การยกเลิกเอกสาร	CHANGE_DOC__ATTR	ยกเลิกเอกสาร
5. การลบ / ทำลายเอกสาร	CHANGE_DOC__ATTR CHANGE_DOC__ATTR CHANGE_DOC__ATTR	การเรียกคืนเอกสาร การลบเอกสาร การทำลายเอกสาร
6. การทำสำเนา	COPY_DOCUMENT	การทำสำเนา
7. การจัดการกับความผิดพลาดของเอกสาร	EXCEPTION_HANDLER	การจัดการกับความผิดพลาดของเอกสาร

ตารางที่ 7.2: สรุปฟังก์ชั่นการทำงานของระบบเอกสารที่มีการลงนาม (ต่อ)

กลุ่มฟังก์ชั่นการทำงานของระบบ	ชื่อฟังก์ชั่น	รายละเอียด
8. การตรวจสอบการแก้ไข	SYSTEM_AUDIT_TRAIT SYSTEM_AUDIT_FIND DOCUMENT_AUDIT_TRAIT DOCUMENT_AUDIT_FIND	บันทึกการแก้ไขเอกสารที่ระบบ บันทึกการแก้ไขเอกสารอ่านและค้นหา บันทึกการแก้ไขเอกสารที่ตัวเอกสาร อ่านและค้นหาบันทึกการแก้ไขเอกสารที่เอกสาร
9. การกำหนดสิทธิ์ผู้ใช้ระบบ	ASSIGN_RIGHT MODIFY_USER MODIFY_GROUP	กำหนดสิทธิ์ผู้ใช้ระบบ สร้าง / แก้ไขผู้ใช้ระบบ สร้าง / แก้ไขกลุ่มผู้ใช้ระบบ
10. การเข้าสู่ระบบ	LOG_ON PASSWORD_CHECK GET_RIGHT	การเข้าสู่ระบบ ตรวจสอบรหัสผ่าน มอบหมายสิทธิ์ให้ผู้ใช้ระบบ

7.4 โครงสร้างแฟ้มข้อมูลถาดเอกสารของผู้ใช้ระบบ

แฟ้มข้อมูลของถาดเอกสารมีไว้เพื่อเก็บข้อมูลว่ามีใครส่งเอกสารใหม่มาให้หรือไม่ โครงสร้างของถาดเอกสารจะใช้รูปแบบเอกซ์เอ็มแอล (XML) ในการจัดเก็บข้อมูล โดยมีโครงสร้างดังต่อไปนี้

```
<INTRAY>
    <INDOC FLAG = NEW | DELETE | PURGE | CANCEL |OPEN>
    <SUBJECT>TOPIC</SUBJECT>
    <FILE_NAME>DOC_NAME<FILE_NAME>
    <FROM>NAME<FROM>
    <TIME>DATE_TIME</TIME>
    <\INDOC>
<\INTRAY>
```

รูปที่ 7.1: โครงสร้างแฟ้มข้อมูลถาดเอกสาร

TOPIC ชื่อหัวข้อของเอกสาร

DOC_NAME ชื่อที่อยู่ของเอกสาร

NAME ชื่อผู้ส่ง

DATE_TIME เวลาที่ส่งรูปแบบ DD/MMM/YYYY HH:MM

FLAG เป็นการกำหนดสถานะของแฟ้มข้อมูลในขณะนั้น

NEW - เป็นเอกสารใหม่ยังไม่ได้เปิด

PURGE - เอกสารที่ถูกลบแล้ว และไม่สามารถเรียกคืนได้ เอกสารโดนลบออกจากระบบ

DELETE - เอกสารที่ถูกลบแล้ว แต่สามารถเรียกคืนมาได้ (Restore)

OPEN - เอกสารถูกเปิดอ่านแล้ว

CANCEL- เอกสารถูกยกเลิก

ระบบต้นแบบที่สร้างขึ้น มีจุดประสงค์เพื่อตรวจสอบการออกแบบโครงสร้างว่ารองรับความต้องการผู้ใช้ได้มากเพียงใด และเมื่อนำมาเขียนโปรแกรมสามารถเขียนได้ยาก/ง่ายอย่างไร ทำให้สามารถนำผลที่ได้ไปปรับปรุงโครงสร้างของเอกสารต่อไป

บทที่ 8

ขั้นตอนการสร้างเอกสารต้นแบบ

เนื่องจากในวิทยานิพนธ์นี้ไม่มีการสร้างระบบที่ใช้ในการสร้างเอกสารต้นฉบับ กล่าวคือ ตัวเอกสารต้นฉบับที่ใช้ในโปรแกรมทดสอบถูกสร้างด้วยการใช้โปรแกรมหลายๆ ตัวมารวมกัน เนื่องจากระบบการสร้างเอกสารอยู่เหนือขอบเขตการวิจัยของวิทยานิพนธ์ฉบับนี้ และเอกสารต้นฉบับที่ใช้จะถูกสร้างเพียงครั้งเดียวและสามารถใช้ได้ตลอดไปจนกว่าจะมีการเปลี่ยนแปลง รูปแบบ โครงสร้างของเอกสารใหม่ ในบทนี้จะกล่าวถึงการสร้างเอกสารต้นแบบเพื่อใช้ในระบบทดสอบการลงนามเอกสารโดยใช้เครื่องมือต่างๆ มาประกอบกัน เช่น โปรแกรมแก้ไขเอกสาร (Text Editor) ชื่อ “XML Editor” ในการแก้ไข / สร้างเอกสารเอกซ์เอ็มแอล (XML)

ขั้นตอนในการสร้างเอกสารสามารถแบ่งเป็นส่วนๆ ได้ ดังนี้

- การสร้างเอกสารหลัก รูปแบบเอกซ์เอ็มแอล
- การสร้างเอกสารกำหนดสิทธิ์ รูปแบบเอกซ์เอ็มแอล
- การสร้างเอกสารกำหนดชุด (version)ของเอกสาร รูปแบบเอกซ์เอ็มแอล
- การสร้างเอกสารกำหนดการลงนามอิเล็กทรอนิกส์ รูปแบบเอกซ์เอ็มแอล
- การสร้างเอกสารกำหนดการแสดงผลแบบเฮชทีเอ็มแอล
- การสร้างโปรแกรมเอเอสพี (ASP) เพื่อใช้ในการควบคุมชุดเอกสารที่สร้างมา

เห็นได้ว่าเอกสารไม่ได้มีเพียงเอกสารเดียว แต่เป็นชุดของเอกสารซึ่งทำงานสัมพันธ์กัน ที่อยู่ของชุดเอกสารโดยปกติจะเก็บอยู่ภายใต้สารบบ (Directory) “document/*Formname*” ภายใต้ “HOME DIRECTORY” ของ Web Server ตัวอย่างเช่น ฟอร์มแบบจ่ายเงิน (Payslip) อยู่ใต้สารบบ (Directory) http://<web server>/document/payslip

สาเหตุที่มีการแยกเอกสารออกเป็นส่วนๆ เพื่อให้สามารถใช้งานเอกสารในลักษณะกระจาย (Distribute) และการแก้ไขเฉพาะส่วนของเอกสารทำได้ง่าย ตัวอย่างเช่น

- ต้องการแก้ไขสิทธิ์ของเอกสาร สามารถแก้ไขในเอกสารกำหนดสิทธิ์เท่านั้น
- สามารถนำเอาเอกสารสิทธิ์ไปใช้กับเอกสารอื่น หรือ เอกสารอื่นสามารถชี้มาที่เอกสารกำหนดสิทธิ์เดียวกันได้ ในกรณีที่ผู้มีอำนาจลงนามมีลักษณะเหมือนกัน
- การแก้ไขรูปแบบฟอร์ม (เปลี่ยนแปลงขนาดตัวอักษรและ ตาราง) สามารถทำได้โดยอิสระจากเนื้อหาของแบบฟอร์ม

8.1 การสร้างเอกสารหลักรูปแบบ XML

จุดประสงค์ของการสร้างเอกสารหลักเพื่อกำหนดโครงสร้างของเอกสารว่ามีส่วนประกอบอะไรบ้าง เช่น “ชื่อบัญชี” “ชื่อแผนก” โดยการสร้างเป็นไปตามรูปแบบของเอกสารเอกซ์เอ็มแอล (XML) การสร้างสามารถใช้โปรแกรมสร้างเอกสารเอกซ์เอ็มแอล (XML Editor) จากไมโครซอฟท์ เป็นตัวสร้าง หรือโปรแกรมที่ใช้สร้างเอกสารเอกซ์เอ็มแอล (XML) อื่น ในเอกสารหลักนี้มีตัวบอกคุณสมบัติ (Attributes) ที่ใช้ในการบอกถึงคุณลักษณะของเอกสารเพิ่มขึ้นมาด้วยกัน 2 ตัว คือ

- SECURE ใช้เป็นตัวชี้ไปยังส่วนกำหนดสิทธิ์ผู้ใช้งานของเอกสาร
- SIGNATURE ใช้เป็นตัวชี้ไปยังส่วนกำหนดการลงลายเซ็นของเอกสาร

รายละเอียดของโครงสร้างตัวชี้ ดูในบทที่ 4 โครงสร้างเอกสาร

8.2 การสร้างเอกสารกำหนดสิทธิ์รูปแบบเอกสาร XML

สามารถสร้างเอกสารกำหนดสิทธิ์ในลักษณะเดียวกับการสร้างเอกสารหลักโดยอิงตามโครงสร้างของเอกสารในบทที่ 4 ในการกำหนดข้อมูลเริ่มต้นเพื่อกำหนดสิทธิ์ สามารถแบ่งได้เป็น 2 ส่วนหลักคือ

1. การกำหนดกลุ่มของผู้มีสิทธิ์ เพื่อความสะดวกในการจัดกลุ่มผู้มีสิทธิ์เหมือนกันไว้ด้วยกัน และเมื่อมีการเปลี่ยนแปลงบุคคลจะเปลี่ยนแปลงที่เดียวแทนที่จะต้องเปลี่ยนแปลงทั้งหมด ในการกำหนดกลุ่มต้องมีการกำหนดชื่อกลุ่ม สมาชิกที่อยู่ในกลุ่ม
2. การกำหนดสิทธิ์ว่าใครมีสิทธิ์อะไรบ้างเพื่อใช้ในการแก้ไขเอกสาร การลงนามเอกสาร ส่วนนี้จะระบุชื่อผู้ใช้ โดยตรงหรือใช้ชื่อกลุ่ม ว่าใครมีสิทธิ์ทำอะไรเมื่อไร และเงื่อนไขที่กำหนดให้ เช่น มีสิทธิ์แก้ไขเมื่อวงเงินน้อยกว่า 100,000

8.3 การสร้างเอกสารกำหนดชุด (version)ของเอกสาร

ในการสร้างเอกสารนี้สร้างเฉพาะโครงสร้างเอกสารตามที่ออกแบบในบทที่ 4 เท่านั้น เนื่องจากโปรแกรมจะทำการแก้ไขเปลี่ยนแปลงเมื่อมีการแก้ไขเอกสารเอง ดังนั้นผู้ใช้จึงเพียงแต่สร้างแฟ้มข้อมูลต้นแบบเท่านั้น โดยที่โครงสร้างเอกสารต้นฉบับจะถูกเก็บไว้ในเอกสารหลัก

8.4 การสร้างเอกสารกำหนดการลงนามอิเล็กทรอนิกส์

จุดประสงค์ของการสร้างเอกสารคือ กำหนดรายละเอียดของการเซ็นโดยมีรายละเอียดในการสร้างดังนี้

1. กำหนดรายละเอียดของขั้นตอนการทำงาน (Algorithm) ในการทำลายเซ็นอิเล็กทรอนิกส์ (Digital Signature) และการทำการแฮชข้อมูล (Hashing) เพื่อเป็นตัวกำหนดมาตรฐานของเอกสารว่าขั้นตอนการทำงาน (Algorithm) ใด ในส่วนนี้ยังทำหน้าที่เก็บกุญแจทั่วไป (Public Key) ของผู้ที่มีสิทธิ์ใช้ / แก้ไขเอกสารด้วย เพื่อการเข้ารหัส / ถอดรหัสข้อมูล รวมถึงชี้ไปยังแฟ้มข้อมูลของภาพถ่ายลายเซ็นที่ใช้แสดงในเอกสาร

2. ส่วนนี้กำหนดชื่อผู้มีอำนาจลงนามว่าต้องการการลงนามเป็นแบบใด เช่น เซ็นร่วมกัน คนเดียว ตามลำดับขั้น ลำดับของการลงนามและเลขรหัสประจำต้นไม้ของลำดับขั้น เพื่อกำหนดลำดับของการลงนาม เช่น เลขกำกับ 2.0 ต้องเซ็นหลัง 1.0 ส่วนเลข 2.2 และ 2.1.มีลำดับเท่ากันใครเซ็นก่อนหลังก็ได้

8.5 การสร้างเอกสารกำหนดการแสดงผล

เนื่องจากการออกแบบได้แยกส่วนการแสดงผลและเนื้อหาของเอกสารออกจากกันทำให้ต้องมีการออกแบบรูปร่างหน้าตาเอกสารว่ามีการจัดวางอย่างไร โดยผู้สร้างเอกสารสามารถใช้โปรแกรมสร้างเอกสารเอชทีเอ็มแอล (HTML Editor) ใดๆ ในการสร้างเอกสาร เอชทีเอ็มแอล (HTML) ขึ้นมา โดยมีข้อกำหนดที่ต้องยึดถือดังต่อไปนี้

- ช่องกรอกข้อมูลต้องใช้กล่องใส่ข้อความ (Text Box) ทุกช่องเป็นตัวรับข้อมูล
- เอกสารทั้งหมดต้องอยู่ภายใต้ฟอร์มอ็อบเจกต์เนื่องจากมีการใช้ฟอร์มในการส่งเอกสาร
- ชื่อของช่องกรอกข้อมูลต้องเหมือนกับแทกในเอกสารหลักเอกซ์เอ็มแอล (XML) เพื่อความสะดวกในการอ้างอิง
- ถ้ามีมากกว่าหนึ่งแถวของข้อมูลให้ใช้หลักการตั้งชื่อว่า XXXN โดย XXX คือ ชื่อของแทกที่ใช้ N คือเลขประจำแถวที่มีค่าตั้งแต่ 1, 2, ...ขึ้นไป
- ชื่อของฟอร์มต้องเหมือนกันในแต่ละเอกสาร คือชื่อว่า “FrontPage_form1” เพื่อให้โปรแกรมที่ใช้อ้างอิงไม่ต้องมาเขียนโปรแกรมใหม่
- ต้องมีการรวมเอาจาวาสคริปต์ (Java Script) แฟ้มข้อมูลในตอนท้ายของเอกสารเพื่อให้แก้ไขเปลี่ยนแปลงข้อมูลในเอกสารได้
- ในส่วนของการลงนามให้ใช้ชื่อ ดังนี้ *sign*_sig คือ ชื่อเขตข้อมูล (field)ที่ใส่รูปลายเซ็น โดย *sign* คือชื่อของผู้ที่มีสิทธิ์เซ็น *sign*_name ชื่อของผู้ลงลายเซ็น *sign*_date วันที่ที่ลงลายเซ็น (ใช้ SPAN ในการกำหนด) โดยรูปเริ่มต้นเป็นรูปว่างเมื่อยังไม่ได้ลงลายเซ็น เช่น ลายเซ็นของผู้ตรวจสอบบัญชี (Audit) คือ audit_sign ชื่อจริงของผู้ตรวจสอบบัญชีคือ audit_name ลงวันที่ audit_date

8.6 การสร้างโปรแกรม ASP

การสร้างโปรแกรมเอเอสพี (ASP) เพื่อใช้ในการควบคุมเอกสาร โปรแกรมนี้คือหัวใจสำคัญของระบบในการควบคุมชุดเอกสารทั้งหมดนี้โดยโปรแกรมนี้จะถูกรวมเข้าไปในชุดเอกสารด้วย

โปรแกรมเอเอสพีเขียนจากภาษาจาวาสคริปต์ (JavaScript) และเอเอสพี หน้าที่หลักของโปรแกรมนี้คือ ทำหน้าที่ติดต่อ แก้ไขข้อมูลกับชุดเอกสารที่มีการลงนาม โปรแกรมถูกออกแบบให้ไม่ขึ้นกับ รูปแบบของเอกสาร และ เขตข้อมูล กล่าวคือ สามารถใช้ได้กับเอกสารทั่วไป ที่มีโครงสร้างข้อมูล และแสดงผลตรงตามที่ได้

กำหนดไว้ในตอนต้นของบทนี้ ในการสร้างสามารถสร้างได้โดย นำแฟ้มข้อมูลแสดงแบบฟอร์มของเอกสารเอชทีเอ็มแอลมาต่อท้ายด้วยโปรแกรมที่ใช้กระทำกับเอกสาร แล้วเก็บแฟ้มข้อมูลข้อมูลเป็นชนิดเอเอสพี จะได้เอเอสพีโปรแกรมที่ใช้ทำกับเอกสารชนิดนั้น

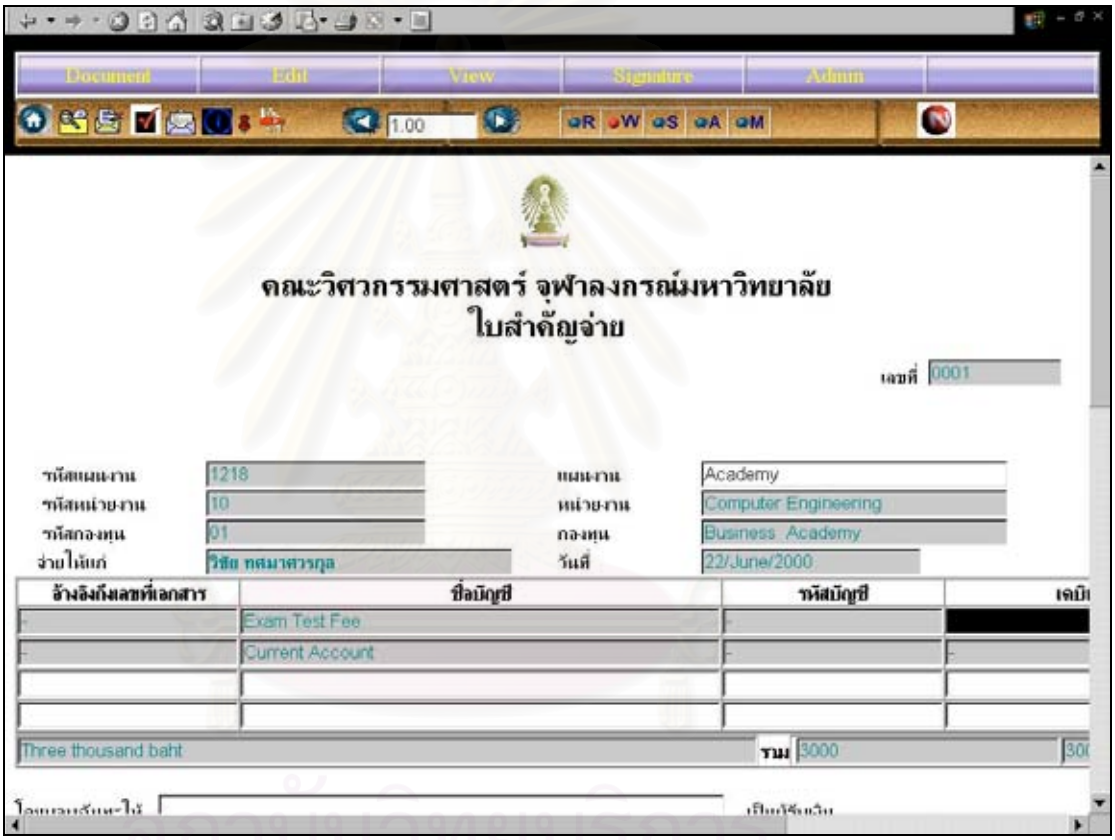
โปรแกรมที่สร้างจะใช้ได้กับชนิดเอกสารนั้นๆ เท่านั้น แต่ว่าจะใช้ได้กับเอกสารทุกฉบับที่เป็นชนิดเดียวกัน กล่าวคือ โปรแกรมเอเอสพีสำหรับใบสั่งจ่ายมีเพียงหนึ่ง และจะใช้ร่วมกันไม่ว่าจะเป็นใบสั่งจ่ายของนาย ก หรือ นาย ข แต่ถ้าต้องการแบบฟอร์มใหม่ต้องสร้างโปรแกรมเอเอสพีขึ้นมาใหม่

ในวิทยานิพนธ์ฉบับนี้ได้ออกแบบแบบฟอร์มในการกรอกข้อมูล เพื่อเป็นแนวในการพัฒนาการสร้างโปรแกรมสร้างเอกสารต่อไป รายละเอียดดูได้จากภาคผนวก

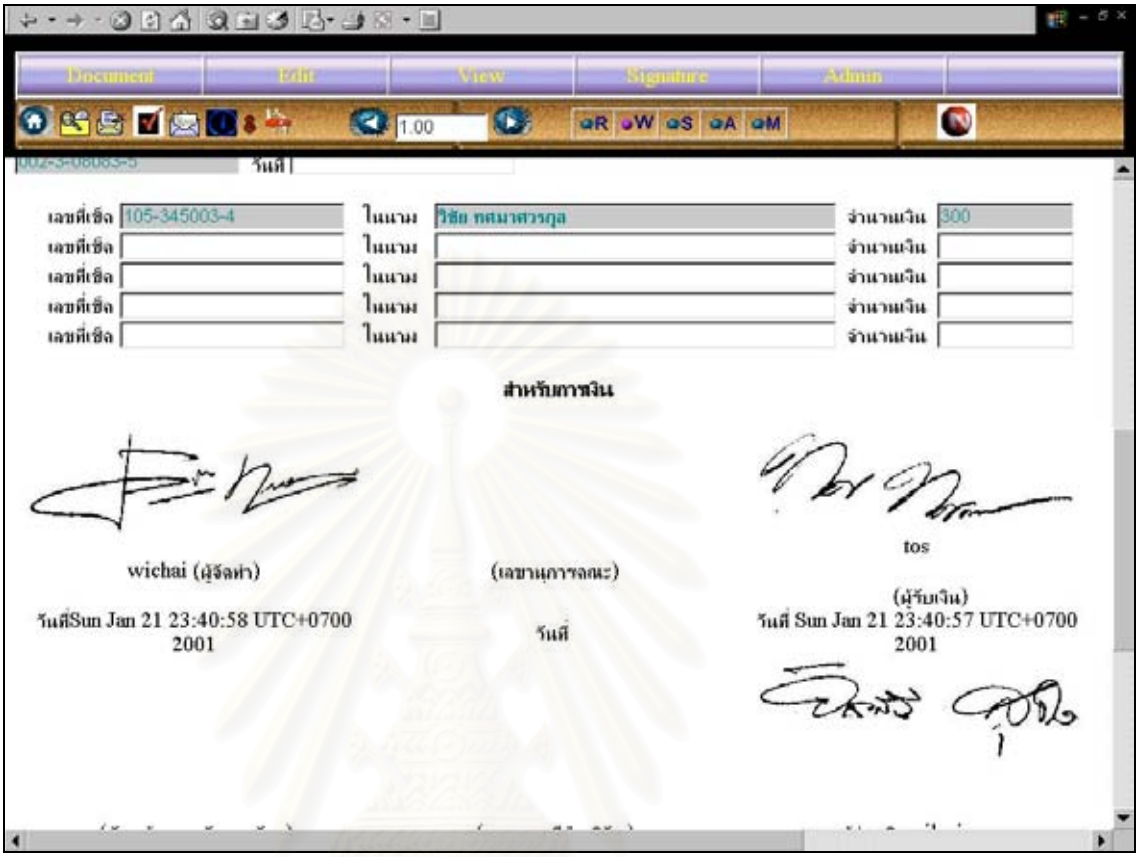
บทที่ 9

ผลการวิจัย

จากการสร้างระบบต้นแบบเพื่อทดสอบโครงสร้างเอกสารที่ออกแบบ พบว่าในส่วนของฟังก์ชั่นหลักสามารถทำงานได้ดี เช่น การกำหนดสิทธิ์ในการอ่าน การแก้ไขของเอกสาร การลงนามและการเรียกดูแต่ละชุด (version) ของเอกสาร ดังตัวอย่างในรูปที่ 9.1



รูปที่ 9.1: แสดงแบบฟอร์มการกรอกข้อมูล



รูปที่ 9.2: แสดงการลงนามในเอกสาร

แต่ในส่วนของฟังก์ชั่นการทำงานย่อยนั้น ระบบต้นแบบยังไม่สามารถทดสอบได้ เช่น การกำหนดการลงลายเซ็นเฉพาะแถว ถึงแม้ได้มีการออกแบบโครงสร้างเอกสารรองรับไว้ แต่การสร้างระบบต้นแบบค่อนข้างยุ่งยาก

เนื่องจากระบบต้นแบบพัฒนาโดยใช้เทคโนโลยีเว็บ ซึ่งทราบกันดีอยู่ว่าออกแบบมาในตอนแรกเพื่อใช้งานการเสนอผลข้อมูลมากกว่าเพื่อใช้ในการพัฒนาโปรแกรม ดังนั้นจึงประสบความยุ่งยากพอสมควรในการพัฒนา เช่น การอ้างอิง / ค้นหา ถึงส่วนต่างๆ ของแบบฟอร์ม ไม่สามารถอ้างอิง / ค้นหาได้ โดยใช้คำสั่งเดียว ทำให้ระบบทำงานช้าลง ถ้าเอกสารมีโครงสร้างใหญ่และกฎของเอกสารซับซ้อนขึ้น

การที่ต้องออกแบบให้ระบบสามารถรองรับเอกสารต่างๆ กัน โดยไม่ขึ้นกับรูปแบบของเอกสาร ค่อนข้างทำได้ลำบาก เนื่องจากเอกซ์เอ็มแอลยังไม่มีมาตรฐานในการรองรับการทำงานที่เป็นแบบฟอร์ม เช่น ใส่ข้อมูลลงในฟอร์มหลักแล้วให้ไปแก้ไขข้อมูลในแฟ้มข้อมูลเอกซ์เอ็มแอล

สำหรับปัญหาที่พบในการทำวิทยานิพนธ์และคิดว่าน่าจะทำการแก้ไขปรับปรุง เพื่อประสิทธิภาพของระบบ คือ

1. เนื่องจากชุดเอกสารมีเอกสารมากกว่า 1 ฉบับในการใช้งาน ดังนั้นทำให้การเขียนโปรแกรมทำได้ลำบาก และการส่งต่อเอกสารจำเป็นต้องส่งให้ทั้งชุดเอกสาร ทำให้ไม่สะดวกในการส่งต่อเอกสาร แต่ผลดีที่ได้คือ เอกสารมีลักษณะกระจาย ทำให้การแก้ไขแก้เพียงที่เดียว แต่ได้ผลหมดทุกชุดเอกสาร เช่น แก้ไขกฎของเอกสารเกี่ยวกับสิทธิ์ของใบสั่งจ่าย สามารถแก้ที่เอกสารกำหนดสิทธิ์ที่เดียว มีผลต่อใบสั่งจ่ายทุกใบที่ถูกสร้างขึ้น ดังนั้นอาจต้องมีการกำหนดเรื่องรูปแบบการส่งของเอกสารว่าทำอย่างไร จึงสามารถทำให้ส่งง่าย เช่น การส่งเป็นสารบบ (Directory)หรือส่งแฟ้มข้อมูลที่บอกว่าเอกสารนี้ประกอบด้วยเอกสารอะไรบ้าง และอยู่ที่ไหน

2. ในการทดสอบใช้เอเอสพีและจาวาสคริปต์ในการทำงาน ซึ่งอาจทำให้ยากต่อการนำไปใช้กับแพลตฟอร์มอื่นๆ เช่น ยูนิกซ์ ควรพิจารณาการออกแบบในลักษณะของวัตถุ COM อันประกอบด้วย คุณสมบัติของเอกสารและวิธีการที่กระทำกับเอกสาร อาจทำให้การใช้งาน / พัฒนากับระบบอื่นทำได้ง่ายขึ้น และไม่จำกัดอยู่กับระบบที่ต้องเป็นเว็บเท่านั้น

3. ระบบยังไม่สามารถรองรับเงื่อนไขที่ซ้ำซ้อนได้ เช่น มีผู้มีอำนาจในการเซ็น 3 คน แต่ในการเซ็นเอกสารจะใช้เพียง 2 คนใน 3 คนนี้ ซึ่งจะเป็น 2 คนใดก็ได้

4. การพัฒนาส่วนแสดงผล (ใช้ เอชทีเอ็มแอล) และส่วนเก็บข้อมูลเอกซ์เอ็มแอลให้ทำงานสัมพันธ์กันทำได้ลำบาก เช่น แก้ไขข้อมูลในฟอร์มเอชทีเอ็มแอล แล้วไปแก้ไขข้อมูลในเอกซ์เอ็มแอล

5. เนื่องจากชุดเอกสารที่ออกแบบเป็นแฟ้มข้อมูลแบบข้อความ จึงควรมีการป้องกันการแก้ไขข้อมูลโดยใช้วิธีต่างๆ เช่น เข้ารหัสแฟ้มข้อมูล กำหนดสิทธิ์การเข้าถึงแฟ้มข้อมูลในระบบปฏิบัติการ (Operating Systems)

บทที่ 10

สรุปและข้อเสนอแนะ

10.1 สรุป

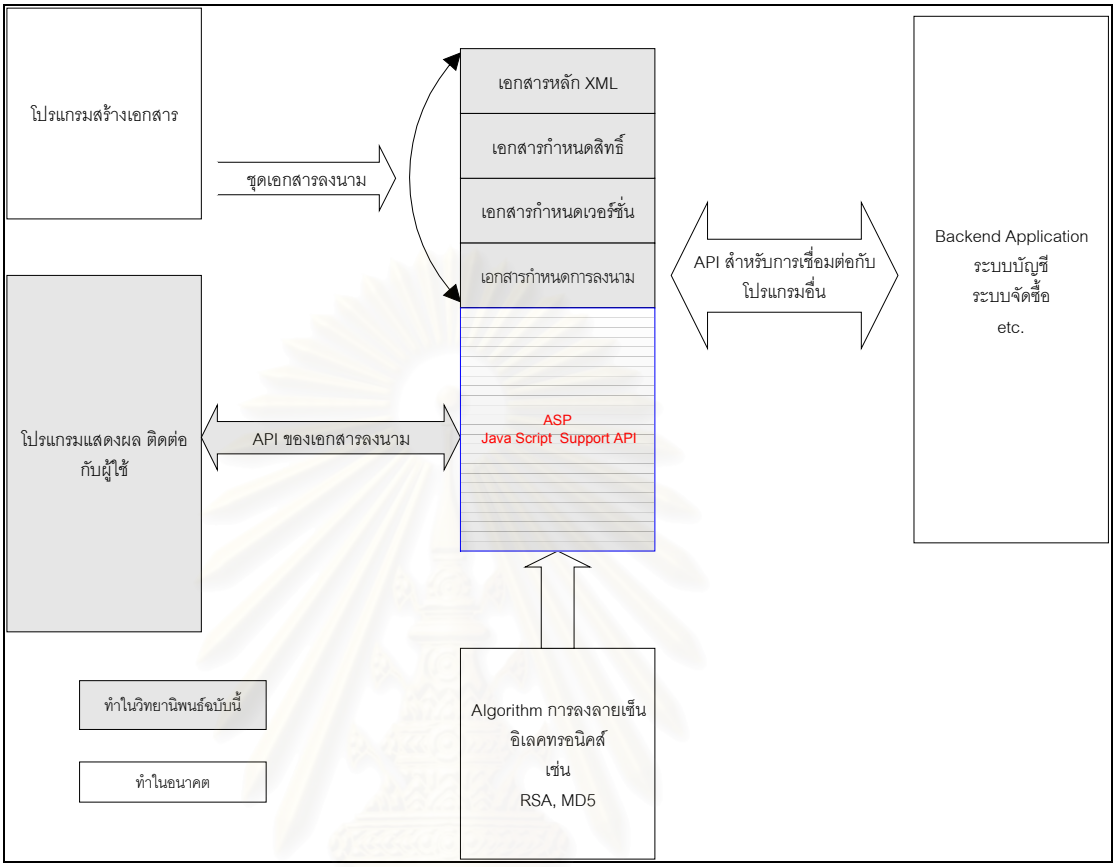
จากผลการวิจัยสามารถนำโครงสร้างเอกสารที่มีการลงนามร่วมกันที่ออกแบบไปเป็นพื้นฐานของโปรแกรมประยุกต์ระบบอื่นๆ ได้ เช่น ระบบการสั่งซื้อ ระบบงานบุคคล (ใบลา ใบประเมินผล) โดยโครงสร้างเอกสารที่ออกแบบจะมีความสมบูรณ์ในตัวเอง กล่าวคือ ข้อมูลของเอกสาร เช่น ใครมีสิทธิ์ใช้เอกสาร ใครมีอำนาจการลงนามแบบไหน (คนเดียว สองคน ลำดับขั้น) ตลอดจนมีการเปลี่ยนแปลงแก้ไขอะไรบ้าง จะถูกเก็บไว้ในโครงสร้างเอกสาร ทำให้สามารถส่งเอกสารไปยังระบบอื่นได้และข้อมูลต่างๆ ยังอยู่ครบ โดยใช้โครงสร้างเอกซ์เอ็มแอล (XML) เป็นตัวกำหนดเนื้อหาของข้อมูล ทำให้หมดปัญหาเรื่องการเก็บข้อมูลในฐานข้อมูล เมื่อต้องการส่งต่อให้ระบบอื่นที่ต้องมีการดึงข้อมูลหลายที่จากฐานข้อมูล และยังยากต่อการพัฒนา เนื่องจากฐานข้อมูลที่ต่างชนิดกัน และวิ่งบนแพลตฟอร์มที่ต่างกัน

ระบบต้นแบบที่ออกแบบยังไม่สามารถรองรับส่วนรายละเอียดของโครงสร้างเอกสารได้ เช่น การกำหนดสิทธิ์ ช่วงเวลาของผู้มีอำนาจในการลงนาม การเซ็นแทน ทั้งนี้ในการใช้งานระบบจริงต้องมีการเขียนฟังก์ชั่นรองรับโครงสร้างเอกสารในส่วนนี้ด้วย และเนื่องจากการสร้างเอกสารต้นแบบต้องทำด้วยมือ จึงทำให้การสร้างเอกสารเพื่อใช้ในระบบต้นแบบค่อนข้างยุ่งยาก และไม่ครอบคลุมโครงสร้างของแบบฟอร์มทั้งหมด

ในภาคผนวกได้เสนอแนะโครงสร้างและหน้าจอของโปรแกรมสร้างเอกสาร เพื่อให้ผู้สนใจสามารถพัฒนาเพิ่มเติมต่อไป

10.2 ข้อเสนอแนะในการพัฒนาระบบ

จากการออกแบบโครงสร้างเอกสารและการทดสอบระบบต้นแบบ พบว่า เอกสารที่ออกแบบยังมีบางส่วนที่ต้องออกแบบเพิ่มเติม เพื่อความสมบูรณ์ในการใช้งานจริง โดยดูได้จากรูปข้างล่างนี้แสดงถึงโครงสร้างรวมของระบบเอกสารที่มีการลงนาม



รูปที่ 10.1: แสดงระบบที่พัฒนาในปัจจุบันและในอนาคต

ในวิทยานิพนธ์ฉบับนี้ได้ทำการออกแบบโครงสร้างของเอกสารที่มีการลงนาม พร้อมทั้งกำหนดฟังก์ชั่นมาตรฐานเพื่อใช้ในการเรียกดูข้อมูล เขียนข้อมูล และการกระทำอื่นๆ กับชุดเอกสาร พร้อมทั้งออกแบบส่วนแสดงผล รับข้อมูล เพื่อส่งไปยังชุดเอกสาร ซึ่งการออกแบบส่วนนี้ขึ้นอยู่กับชนิดของโปรแกรมที่ผู้ใช้ต้องการเป็นหลัก

ในส่วนที่ยังไม่ได้พัฒนา แต่คิดว่าควรมีการพัฒนาเพิ่มเติมคือ ส่วนการสร้างโปรแกรมสร้างเอกสาร เนื่องจากโครงสร้างของเอกสารมีความซับซ้อน ไม่เหมาะกับผู้ใช้งานที่ไม่มีความรู้เรื่องเอกซ์เอ็มแอล จึงควรมีโปรแกรมสร้างเอกสารที่ง่ายต่อการใช้งานเพื่อให้ผู้ใช้สามารถออกแบบเอกสารได้ง่ายยิ่งขึ้น

อีกส่วนที่สำคัญคือ การพัฒนาระบบให้เชื่อมต่อกับงานระบบอื่นที่ใช้อยู่ในองค์กร ตัวอย่างเช่น ระบบบัญชี ระบบจัดซื้อ ระบบงานบุคคล จากเดิมคือ ผู้ใช้จะนำข้อมูลที่กรอกในแบบฟอร์มแล้วป้อนเข้าคอมพิวเตอร์ เช่น รับใบสั่งซื้อมาแล้วป้อนคำสั่งซื้อที่ผ่านการอนุมัติแล้ว (ตรวจสอบจากลายเซ็น) เข้าสู่ระบบ ในอนาคตสามารถเชื่อมระบบเหล่านี้กับระบบส่วนหน้า (Front End) ที่ใช้เอกสารเป็นตัวกลาง ทำให้ไม่ต้องใช้กระดาษอีก

ต่อไป เมื่อการอนุมัติครบถ้วนตามกฎของเอกสาร เอกสารพร้อมข้อมูลจะถูกป้อนเข้าสู่โปรแกรมประยุกต์อื่นต่อไป ทำให้เป็นการใช้งานคอมพิวเตอร์เต็มรูปแบบ ตั้งแต่เริ่มจนจบ

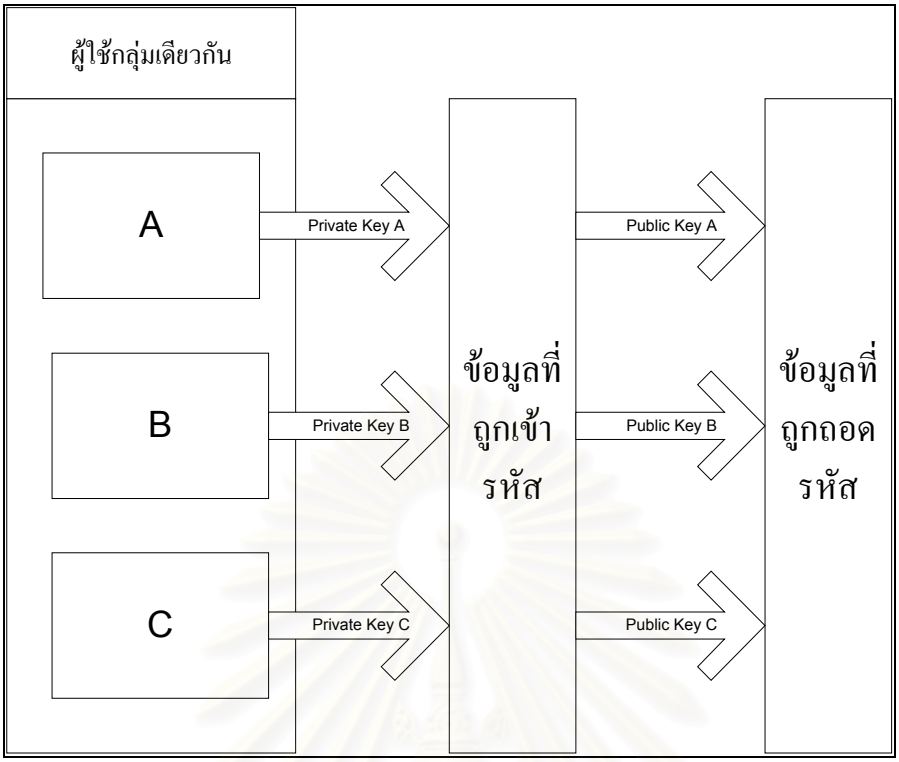
ในส่วนนี้จำเป็นต้องมีการออกแบบการเชื่อมโยงโปรแกรม (Application Programming Interface หรือ API) สำหรับเชื่อมต่อกับโปรแกรมประยุกต์อื่น ตัวอย่างเช่น

- การเชื่อมโยงโปรแกรม (API) ด้านการติดต่อกับฐานข้อมูลของโปรแกรมประยุกต์
- การเชื่อมโยงโปรแกรม (API) ด้านการดึงข้อมูลและข้อมูลการตรวจสอบลายเซ็น
- การเชื่อมโยงโปรแกรม (API) ด้านการกระทำกับชุดเอกสารที่มีการลงนามเป็นจำนวนมาก (Transaction)
- การเชื่อมโยงโปรแกรม (API) ในการเปลี่ยนแปลงค่าข้อมูล (Convert Data) จากรูปแบบหนึ่งไปอีกรูปแบบหนึ่ง

10.3 ข้อเสนอแนะในการเข้ารหัสข้อมูลและความปลอดภัยของเอกสาร

ในส่วนการเข้ารหัสเพื่อการลงลายเซ็นได้มีการกำหนดโครงสร้างเอกสารให้สามารถระบุขั้นตอนวิธี (Algorithm) ในการลงนามอิเล็กทรอนิกส์ได้ แต่จากการออกแบบระบบต้นแบบและการทดสอบระบบทำให้เกิดข้อสังเกตด้านความปลอดภัยของเอกสาร ดังต่อไปนี้คือ

1. ขั้นตอนวิธีการเข้ารหัสที่ใช้ต้องให้ผลลัพธ์เป็นรหัสแอสกี (ASCII) โดยเป็นตัวอักษรเท่านั้นและไม่รวมอักษรพิเศษ เช่น > < “ เพื่อหลีกเลี่ยงปัญหาในการตีความโดยตัวแจง (Parser) ของเอกสารเอกซ์เอ็มแอล
2. เนื่องจากการเข้ารหัสข้อมูลจะกระทำเฉพาะข้อมูลเท่านั้น ไม่รวมถึงการเข้ารหัสโครงสร้างเอกสารทั้งเอกสาร ทำให้เกิดโอกาสที่ผู้อื่นมาแก้ไขโครงสร้างเอกสาร เพื่อเปลี่ยนแปลงสิทธิ์และการลงนามของเอกสาร ยกตัวอย่างเช่น ในระบบต้นแบบใช้การตรวจสอบจากผู้ใช้ที่บันทึกเข้า (login) ระบบ ทำให้ผู้ดูแลระบบ (System Administrator) มีสิทธิ์แก้ไขเอกสารได้ ถึงแม้ว่าจะไม่มีสิทธิ์ในเอกสารก็ตาม จึงต้องมีการกำหนดสิทธิ์ในรายละเอียดว่าใครมีสิทธิ์แก้ไขเอกสาร และใครมีสิทธิ์แก้ไขโครงสร้างเอกสาร ซึ่งอาจจะเป็นกลุ่มเดียวกันหรือว่าต่างกลุ่มกันก็ได้ และต้องมีการเข้ารหัสโครงสร้างเอกสารเพื่อป้องกันผู้ไม่มีสิทธิ์แก้ไขและดูได้
3. การกำหนดสิทธิ์การลงนามในลักษณะบทบาททำให้มีปัญหาในด้านการใช้กุญแจเข้ารหัสข้อมูล เนื่องจากอาจมีได้มากกว่า 1 คน ใน 1 บทบาท เมื่อมีการลงนามโดยใช้กุญแจส่วนตัวก็จะแตกต่างกัน ดังนั้นเวลาถอดรหัสก็ต้องใช้กุญแจที่แตกต่างกันในการถอดรหัส ทำให้ต้องถอดรหัสโดยใช้กุญแจที่แตกต่างกัน เนื่องจากในปัจจุบันยังไม่มีเทคโนโลยีของกุญแจกลุ่มซึ่งสามารถเป็นได้ทั้งกุญแจของส่วนตัว และกุญแจของกลุ่ม รูปข้างล่างแสดงถึงการใช้กุญแจที่แตกต่างกันในการเข้า/ถอดรหัส



รูปที่ 10.1: แสดงการใช้กุญแจในการเข้ารหัสที่ต่างกันในผู้ใช้กลุ่มเดียวกัน

4. ระบบต้นแบบใช้การเข้าสู่ระบบ (login) ในการระบุตัวผู้ใช้ แต่ในทางปฏิบัติที่ชุดเอกสารต้องมีการส่งต่อให้ระบบอื่น จะต้องมีขั้นตอนในการระบุตัวผู้ใช้ที่สามารถส่งไปพร้อมกับเอกสารได้โดยไม่ติดกับระบบ ตัวอย่างเช่น ในชีวิตประจำวันเราใช้บัตรประชาชนในการตรวจสอบบุคคลโดยไม่จำเป็นต้องไปดึงข้อมูลจากฐานข้อมูล ในโครงสร้างเอกสารได้กำหนดแทก <PUBLIC_KEY> โดยเราสามารถตรวจสอบผู้ใช้ได้โดยการตรวจสอบกุญแจสาธารณะของคนนั้นกับหน่วยงานที่ตรวจสอบการให้กุญแจ (Certificate Authority) ว่าตรงกันหรือไม่

10.4 ข้อเสนอแนะด้านการจัดเก็บแฟ้มข้อมูล

จากการทดสอบพบว่าถ้ามีชุดเอกสารจำนวนมาก จำนวนแฟ้มข้อมูลจะมากขึ้นตาม เนื่องจากหนึ่งชุดเอกสารมีแฟ้มข้อมูลย่อย 4 เอกสาร ถึงแม้ว่าเอกสารกำหนดสิทธิ์ เอกสารหลัก (โครงสร้างเอกสาร) และเอกสารกำหนดการลงนามสามารถใช้ร่วมกันได้ แต่จำนวนเอกสารก็จะมีมากขึ้นถ้ามีการใช้งานมากขึ้น

ในการพัฒนาได้มีการกำหนดชื่อของแฟ้มข้อมูล (Filename Convention) เพื่อให้สะดวกในการอ้างอิง ดังต่อไปนี้

เอกสารกำหนดสิทธิ์ มีชื่อเป็น *DOC_NAME*_SECURE.XML

เอกสารกำหนดโครงสร้าง	มีชื่อเป็น	*DOC_NAME*.XML
เอกสารกำหนดการลงนาม	มีชื่อเป็น	*DOC_NAME*_SIGN.XML
เอกสารกำหนดชุด (version)ของเอกสาร	มีชื่อเป็น	*DOC_NAME*_VERSION.XML

โดย *DOC_NAME* คือชื่อของเอกสาร

เอกสารที่เป็นชนิดเดียวกันสามารถใช้เอกสารกำหนดการลงนาม เอกสารกำหนดสิทธิ์และเอกสารกำหนดโครงสร้างร่วมกันได้ มีเพียงเอกสารกำหนดชุดที่จะแตกต่างกันไประหว่างแต่ละเอกสาร ดังนั้นระบบที่พัฒนาจะต้องสามารถที่จะอ้างอิงและสืบค้นเอกสารเหล่านี้ได้ โดยมีการทำสารบบ (Directory) ซึ่งอาจจะใช้ฐานข้อมูลเป็นตัวจัดการกับชุดเอกสารเพื่อการสืบค้นและค้นหาเอกสารได้รวดเร็วขึ้น

รายการอ้างอิง

ภาษาอังกฤษ

[1] Amoroso, E. G. Fundamental of Computer Security Technology. New Jersey: Prentice Hall PTR, 1994.

[2] Bernard, R. The Corporate Intranet. New York: John Wiley & Sons, 1996.

[3] Bernstein, T., et. al. Internet Security for Business. New York: John Wiley & Sons, 1996.

[4] Bob, Reselman Active Server Page 3.0 By Example. Indiana: QUE, 2000.

[5] Dynamic HTML Reference and Software Development Kit. Washington: Microsoft Press, 1999.

[6] Floyd, M. Building Web Sites with XML. New Jersey: Prentice Hall PTR, 2000.

[7] Ford, W.; Baum, M. S. Secure Electronic Commerce. New Jersey: Prentice Hall PTR, 1997.

[8] Goldfarb, C. F., Prescod P. The XML Handbook. New Jersey: Prentice Hall PTR, 1998.

[9] Goldfarb, C. F. Designing XML Internet Applications. New Jersey: Prentice Hall PTR, 1998.

[10] Kravitz, Jeff. SDML- Signed Document Markup Language. W3C Note, Financial Service Technology Consortium (June 1998).

[11] Marchal, B. XML By Example. Indiana: QUE, 2000.

[12] Pfleeger, C. P. Security in Computing. Second Edition. New Jersey: Prentice Hall PTR, 1997.

[13] Schneier, B. Applied Cryptography. Second Edition. New York: John Wiley & Sons, 1996.

[14] Schurman, E. M.; Pardi W. J. Dynamic HTML in Action. Second Edition. Washington: Microsoft Press, 1999.

[15] Simpson, J. E. Just XML. New Jersey: Prentice Hall PTR, 1999.

[16] Spencer, P. Professional XML Design and Implementation. Birmingham: Wrox Press, 1999.

[17] Stallings, W. Protect Your Privacy, The PGP User's Guide. New Jersey: Prentice Hall PTR, 1995.

[18] Vitali, F.; Durand, D. G. Using Versioning to Provide Collaboration. **WWW** Fourth International World Wide Web Conference Proceedings (December 1995)

[19] Weissinger, K. A. ASP In a Nutshell. Second Edition. California: O'Reilly & Associates, 2000.

[20] www.rsa.com. Securing Communications on the Intranet and over the Internet. RSA Encrytion Standard. Available from:http://home.fr.netscape.com/newsref/ref/128bit.เอชทีเอ็มแอล

ภาคผนวก

ภาคผนวก ก.

การเปลี่ยนข้อมูลจากฟอร์มกระดาษเป็นอิเล็กทรอนิกส์

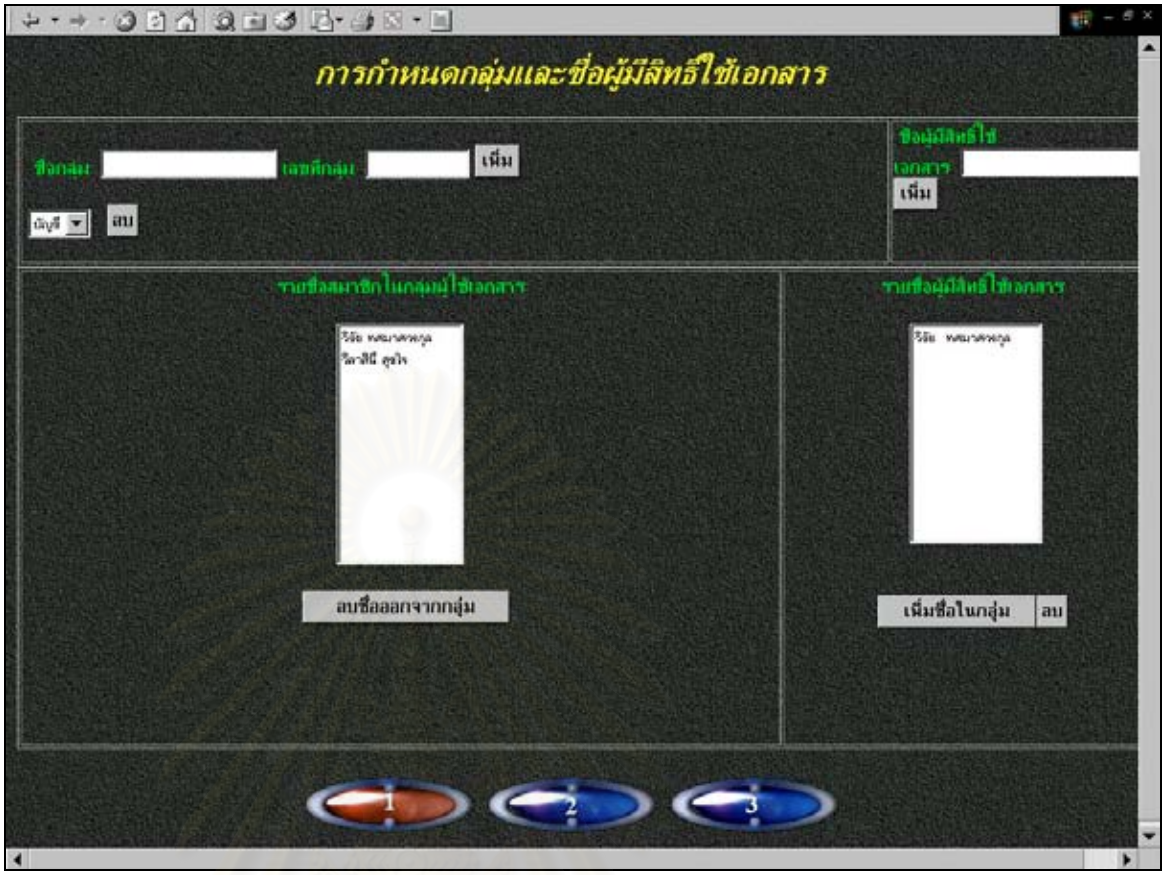
ในการสร้างเอกสารอิเล็กทรอนิกส์ การที่จะให้ผู้ใช้มาสร้างเอกสาร XML คงทำได้โดยลำบาก เนื่องจากผู้ใช้ไม่ทราบโครงสร้างเอกสาร XML หรือโครงสร้างของเอกสารที่มีการลงนาม ในทางกลับกัน ถ้าให้ผู้พัฒนามาทำการออกแบบโครงสร้างเอกสาร XML จะเกิดปัญหาว่าผู้ออกแบบไม่ทราบว่า เขตข้อมูล (field)แต่ละเขตข้อมูล (field)ในหนึ่งฟอร์มคืออะไร และใครมีสิทธิ์ในการแก้ไขเขตข้อมูล (field)ใดบ้าง รวมทั้งอำนาจหน้าที่ในการลงนาม ลำดับขั้นของการลงนาม

ดังนั้นจึงได้ออกแบบฟอร์มในการถามผู้ใช้ระบบ เพื่อให้ผู้ใช้ระบบกรอกข้อมูล รายละเอียดสิทธิ์ของผู้ใช้ที่กระทำกับฟอร์มนี้ หลังจากกรอกเสร็จแล้ว ผู้พัฒนาสามารถนำข้อมูลที่ได้เพื่อไปออกแบบเอกสาร XML ต่อไป หรือส่งต่อไปยังโปรแกรมการสร้างเอกสาร (EDITOR) เพื่อทำการสร้างเอกสาร XML โดยอัตโนมัติ

โดยแบบฟอร์มแบ่งออกเป็น 3 ส่วนใหญ่ๆคือ

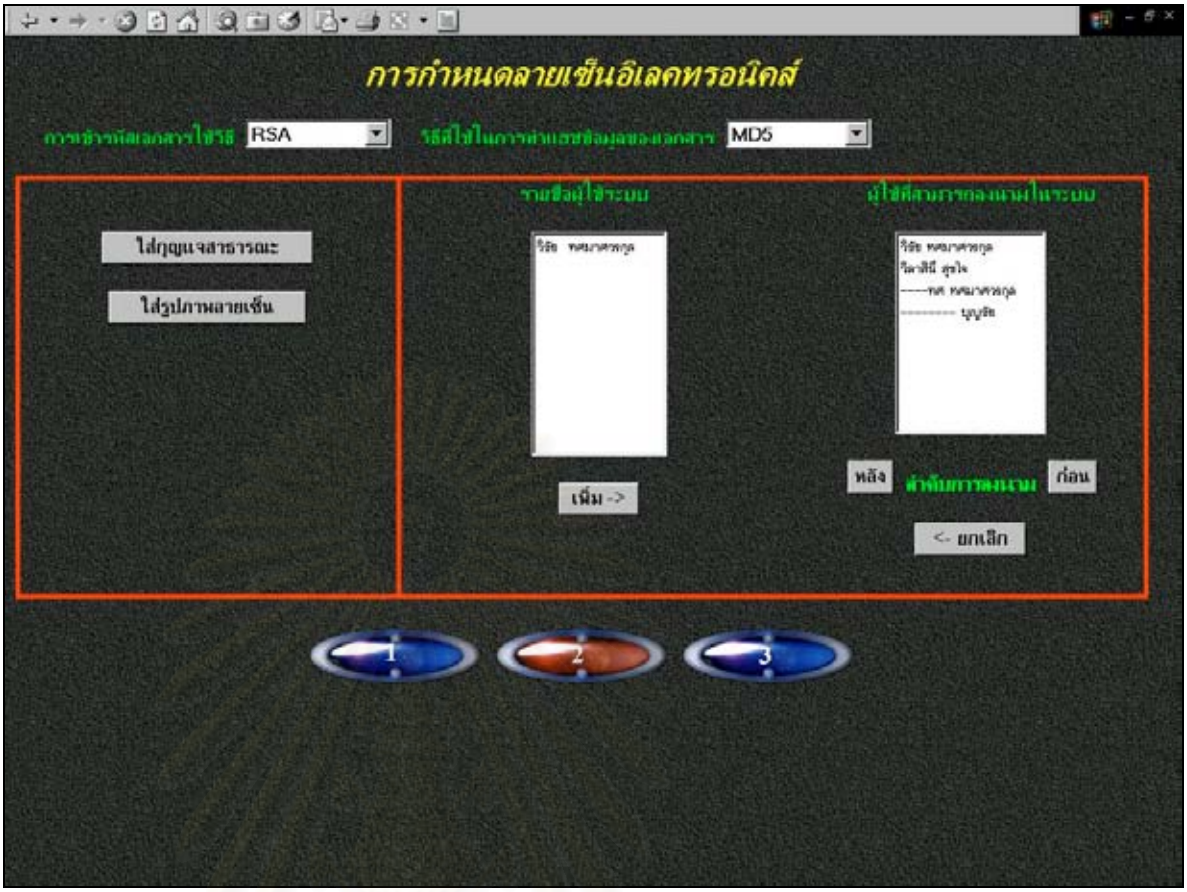
- ส่วนกำหนดกลุ่มของสิทธิ์ เพื่อการจัดกลุ่มของผู้มีสิทธิ์เหมือนกัน และการเพิ่มชื่อผู้มีสิทธิ์เข้าไปในระบบ
- ส่วนการกำหนดเขตข้อมูล (field) สิทธิ์ของผู้ใช้ ใช้ในการกำหนดชื่อเขตข้อมูล (field) สิทธิ์ของผู้ใช้ ว่าใครมีสิทธิ์ทำอะไรบ้างในเขตข้อมูล (field)นี้
- ส่วนกำหนดผู้มีสิทธิ์ในการลงนามเอกสาร เพื่อการกำหนดผู้มีสิทธิ์ลงนามในเอกสาร รวมทั้งข้อมูลเกี่ยวกับการลงนาม เช่น กุญแจสาธารณะ (Public Key) รูปลายเซ็น

ข้อมูลที่ได้จากแบบฟอร์มช่วยในการสร้างเอกสารในรูปแบบ XML โดยข้อมูลที่ได้อาจนำส่งต่อให้โปรแกรมสร้างเอกสาร (EDITOR) หรือให้ผู้ดูแลระบบในการดำเนินการสร้างเอกสารตามข้อมูลของแบบฟอร์มที่ให้ไว้



รูปที่ 1: แสดงหน้าจอการกำหนดชื่อผู้มีสิทธิ์การใช้เอกสาร

หน้าจอนี้ทำหน้าที่ในการกำหนดชื่อผู้มีสิทธิ์ใช้งานเอกสาร โดยมีการสร้าง 2 ประเภทคือ การสร้างชื่อผู้ใช้แต่ละคน และการสร้างในลักษณะของกลุ่มของผู้ใช้ ทั้งนี้ เพื่อความสะดวกในการเปลี่ยนสมาชิกภายหลัง โดยไม่มีผลกระทบต่อตัวเอกสาร

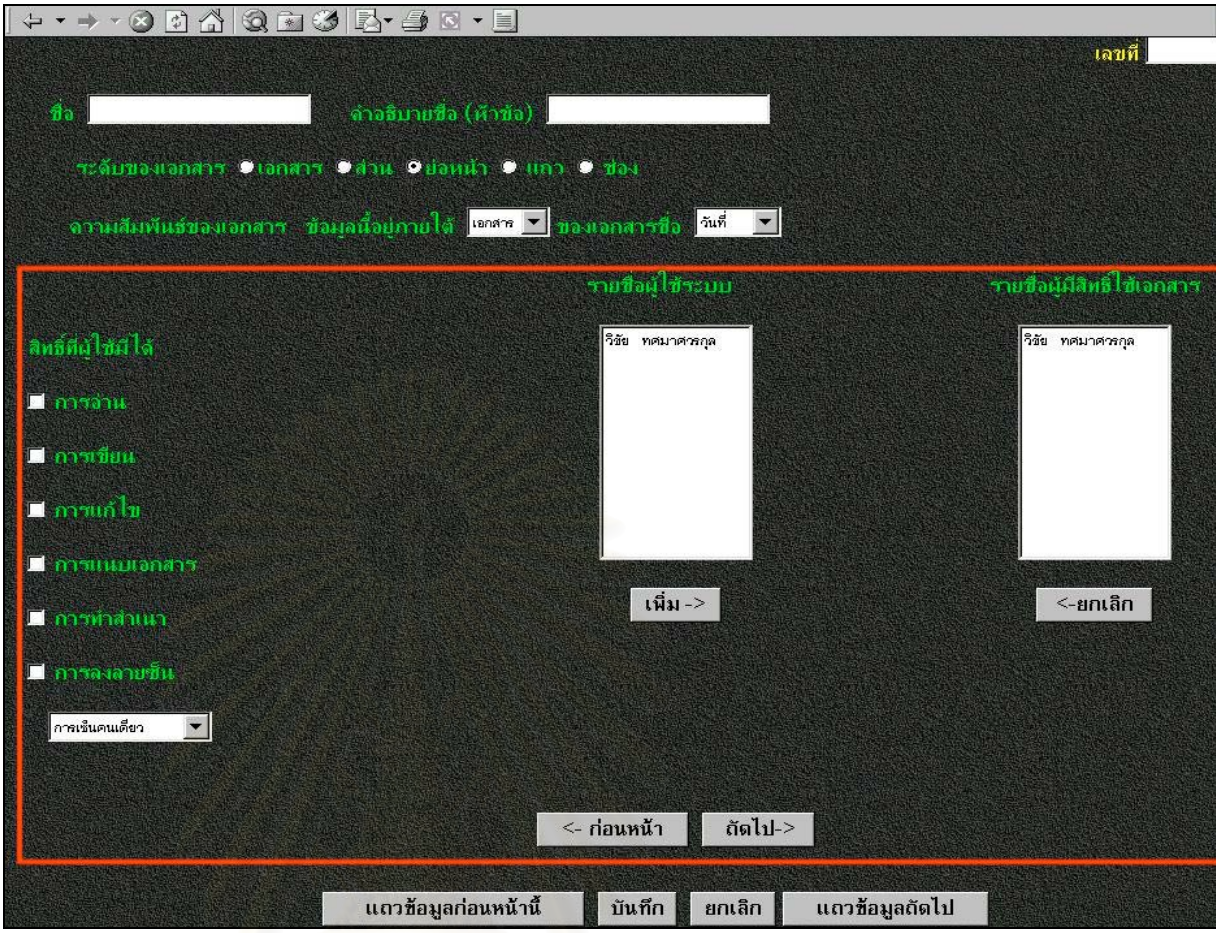


รูปที่ 2: แสดงหน้าจอการกำหนดชื่อผู้มีสิทธิ์ลงลายเซ็นอิเล็กทรอนิกส์

การกำหนดลายเซ็น หน้าจอนี้ทำหน้าที่ในการกำหนดผู้มีอำนาจในการลงนาม ว่าใครมีสิทธิ์ลงนามในเอกสารบ้าง โดยแบ่งเป็นสองส่วน คือ

1. ส่วนการกำหนดการเข้ารหัสเอกสาร เช่น RSA, SHAR และสิทธิ์ในการทำแฮชซิ่งข้อมูล (Hashing) เพื่อกำหนดขบวนการการลงนามอิเล็กทรอนิกส์

2. ส่วนการกำหนดผู้ลงนามในเอกสาร คือผู้ที่มีสิทธิ์ลงนามในเอกสาร ในส่วนนี้ต้องมีการใส่ข้อมูลของผู้ใช้ในการลงนามอิเล็กทรอนิกส์ด้วย เช่น การใส่กุญแจสาธารณะ (Public Key) การใส่รูปภาพลายเซ็น นอกจากนี้ยังมีการกำหนดผู้ที่มีอำนาจลงนามก่อนหลังได้ โดยการเลือกชื่อผู้ลงนาม และกำหนดการลงนามก่อนหรือหลัง โดยชื่อที่ต้องลงนามทีหลังจะอยู่ริมขวาสุด ชื่อที่ต้องลงนามก่อนจะอยู่ถัดไปทางซ้ายตามลำดับดังรูป ส่วนถ้าลงนามร่วมกัน 2 คน จะอยู่ที่ตำแหน่งเดียวกัน หมายถึง ใครลงก่อนหรือหลัง ก็ได้



รูปที่ 3: แสดงหน้าจอการกำหนดโครงสร้างเอกสาร

หน้าจอนี้กำหนดโครงสร้างของเอกสาร โดยเป็นการกำหนดส่วนต่างๆ ที่สำคัญของเอกสาร และการแบ่งเอกสารออกเป็นส่วนๆ (ได้แก่ เอกสาร ส่วน ย่อหน้า แถว ช่อง) โดยผู้ใช้ต้องบอกระบบเกี่ยวกับคุณสมบัติของเอกสาร เช่น สิทธิ์การใช้งาน ใครมีสิทธิ์ใช้ในส่วนนี้ และส่วนนี้สัมพันธ์กับส่วนอื่นอย่างไร ชนิดของการเซ็น (เซ็นคนเดียว ลำดับขั้น) ผู้สร้างต้องกรอกรายละเอียดข้อมูลของแบบฟอร์มทีละส่วนจนจบเอกสาร

โปรแกรมสร้างเอกสาร (Editor) ควรนำข้อมูลที่ได้จากแบบฟอร์มทั้ง 3 เพื่อสร้างชุดเอกสารรูปแบบ XML ทั้ง 4 เอกสาร อันได้แก่ เอกสารหลัก เอกสารกำหนดชุด (version) เอกสารกำหนดสิทธิ์ และ เอกสารกำหนดการลงนาม รวมทั้งสร้างฟอร์ม เอชทีเอ็มแอลอย่างง่ายขึ้น อันประกอบด้วย เขตข้อมูล (field)ข้อมูลต่างๆ ตาราง ผู้ใช้สามารถนำเอกสารเอชทีเอ็มแอล นี้ไปออกแบบจัดรูปแบบใหม่ โดยใช้ โปแกรมสร้างเอชทีเอ็มแอลที่มีอยู่ในตลาด หลังจากนั้นจึงทำการสร้างโปรแกรม ASP จากรูปแบบฟอร์ม เอชทีเอ็มแอล ตามที่ได้กล่าวไว้ในตอนต้น ซึ่งโปรแกรมสร้างเอกสารควรสามารถทำหน้าที่เหล่านี้ได้โดยอัตโนมัติ

ประวัติผู้วิจัย

วิชัย ทศมาศวรกุล จบการศึกษาจากคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า ลาดกระบัง สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์ เมื่อปี พ.ศ. 2531 ขณะทำวิทยานิพนธ์ (พ.ศ. 2543) ได้ทำงานอยู่ที่ บริษัท ฮิวเลตต์ – แพคการ์ด (ประเทศไทย) จำกัด ในตำแหน่งวิศวกรที่ปรึกษาด้านเทคนิค ในส่วนของการแก้ไขโปรแกรมประยุกต์ต่างๆ ให้สามารถใช้งานได้กับภาษาท้องถิ่นในเอเชีย เช่น ญี่ปุ่น เกาหลี จีน ไทย

ความสนใจส่วนตัวอยู่ที่เทคโนโลยีอินเทอร์เน็ต เวิร์คโฟลว์โปรแกรม การโปรแกรมหุ่นยนต์โดยใช้ไมโครโปรเซสเซอร์ Basic Stamp เวลาว่างใช้เวลาอยู่กับครอบครัว และ ท่องเที่ยว

E-mail : wichai@thaiinvent.com

http : www.geocities.com/wichai_tos