



PROJECT NO. 67

ระบบจัดเก็บและจัดการเอกสารภายในห้องบรรณสารสนเทศ

MR.AKARAPON BOONSERMSAKUL

MS.THANAPORN PITIANUSORN

MR.ANNOP KONGSOMBATCHAROEN

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR

THE DEGREE OF BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

FACULTY OF ENGINEERING

KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI

2020

Project No. 67

ระบบจัดเก็บและจัดการเอกสารภายในห้องบรรณสารสนเทศ

Mr.Akarapon Boonsermsakul

Ms.Thanaporn Pitianusorn

Mr.Annop Kongsombatcharoen

A Project Submitted in Partial Fulfillment

of the Requirements for

the Degree of Bachelor of Engineering (Computer Engineering)

Faculty of Engineering

King Mongkut's University of Technology Thonburi

2020

Project Committee

.....

Project Advisor

(Asst.Prof. Suthathip Manee, Ph.D.)

.....

Committee Member

(Dr.Prapong Prechaprapraranwong, Ph.D.)

.....

Committee Member

(Asst.Prof.Sanan Srakaew)

.....

Committee Member

(Asst.Prof.Surapont Toomnark)

Project Title	Project No. 67 ระบบจัดเก็บและจัดการเอกสารภายในห้องสมุดสารสนเทศ
Credits	3
Member(s)	Mr.Akarapon Boonsermsakul Ms.Thanaporn Pitianusorn Mr.Annop Kongsombatcharoen
Project Advisor	Asst.Prof. Suthathip Manee, Ph.D.
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2020

### **Abstract**

KMUTT's library have collected the archive of valued documents. Because these document have not transformed into digital form, there is vital problem in searching for information in these document for librarian and patrons. In this project, we developed web platform to digitize these document into digital format and implement the search function that facilitate the librarian and patron to search for information. The platform consists of 2 components. The first part is importing documents and digitization. In this step, we applied image processing techniques such as Morphology Transformation to preprocess the images of documents and transform the images to full text data by using Tesseract. After getting the text files, we tokenize the text into words by using the Deepcut library and find the significant words of the document by using the TF-IDF algorithm. In the second part, we start by getting the input from the user and use the word2Vec model to find a similar word. And take input and similar words to get the TF-IDF score that we generate at first to find the best document for the input word. For comparison correct of OCR is 74.75% , use OCR and correction is 76.61%.

**Keywords:** Natural language processing / RESTful Service / Optical character recognition / Image Processing / Information retrieval / Term Frequency-Inverse Document Frequency / Word2Vec / Word Embedded

หัวข้อปริญญาในพนธ์	ระบบจัดเก็บและจัดการเอกสารภายในห้องรับรองสารสนเทศ KMUTT Archives Management Platform
หน่วยกิต	3
ผู้เขียน	นายอัครพล บุญเสริมศักดิ์กุล นางสาวอรอนพร ปิติอนุสรณ์ นายอรรถพง กองสมบัติเจริญ
อาจารย์ที่ปรึกษา	ผศ.ดร.สุราทิพย์ มณีวงศ์วัฒนา
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2563

### บทคัดย่อ

การจะสืบค้นข้อมูลจากเอกสารหรือขั้นหนังสือที่มีการรวบรวมข้อมูลไว้ตั้งแต่อดีตจนเป็น ปัจจุบันอย่างหนึ่งของเจ้าหน้าที่บรรณาธิการที่ต้องทำการดูแลเอกสารเหล่านี้ เนื่องจาก การที่ยังไม่มีการเก็บหนังสือและเอกสารไว้อยู่ในรูปแบบของข้อมูลดิจิตอลทำให้ต้อง สืบค้นโดยการค้นหาเอกสารและหนังสือแต่ละเล่มโดยการดูจากเนื้อหาสารบัญเพื่อให้ได้หนังสือที่ตรงกับข้อมูลที่ต้องการมากที่สุด ซึ่งการที่ค้นหาจากหน้าสารบัญของ หนังสือแต่ละเล่มก็จะทำให้การค้นหาเป็นไปอย่างล่าช้า และบางครั้งการดูเพียง แค่สารบัญก็อาจจะทำให้ได้หนังสือที่ไม่ตรงกับความต้องการของผู้ที่เข้ามายืนหนังสือ ในโครงการนี้เราได้ทำการพัฒนาการระบบจัดเก็บและค้นหาเอกสารอิเล็กทรอนิกส์ โดยแบ่งออกเป็น 2 ขั้นตอนคือ การนำเข้าข้อมูล และการสร้างระบบค้นหา โดยขั้นตอนการนำเข้าข้อมูล เราจะเริ่มจากการเตรียมข้อมูลรูปภาพ เพื่อเตรียมข้อมูลรูปภาพที่ได้มา ก่อนจะนำไปผ่านกระบวนการ OCR เพื่อแปลงรูปภาพเหล่านี้ให้อยู่ในรูปของข้อมูลดิจิตอล โดยการเก็บข้อมูลในรูปแบบของ Information Retrieval เพื่อช่วยให้ความเร็วการค้นหาไม่มีประสิทธิภาพมากยิ่งขึ้น และนำข้อมูลมาทำการตัดคำ และเช็คคำผิด จากนั้นจะนำมาหาคำสำคัญของหนังสือหรือเอกสารนั้น ๆ โดยการใช้การหาค่าคะแนนแบบ TF-IDF ส่วนการสร้างระบบการค้นหา จะเริ่มจากการรับคำค้นมาจากผู้ใช้และทำการนำคำที่ได้ไปเข้าโมเดล word2Vec เพื่อหาคำที่ใกล้เคียง จากนั้นนำคำใกล้เคียงและคำค้นหาไปดึงคะแนน TF-IDF ที่เก็บไว้เพื่อค้นหาว่า มีเอกสารหรือหนังสือแผ่นไหนที่มีคะแนนที่ต้องแรงและใกล้เคียงกับคำค้นหามากที่สุด โดยผลลัพธ์จากการทำ OCR ถูกต้อง 74.75 % และเมื่อนำมาผ่านกระบวนการแก้คำผิดได้ความถูกต้องอยู่ที่ 76.61%

**คำสำคัญ:** Natural language processing / RESTful Service / Optical character recognition / Image Processing / Information retrieval / Term Frequency-Inverse Document Frequency / Word2Vec / Word Embedded

## กิตติกรรมประกาศ

ขอขอบคุณนางสาวอารยา ศรีบัวบาน เจ้าหน้าที่หอบรรณสารสนเทศและ พศ.ดร.สุราทิพย์ มนิวงศ์วัฒนา อาจารย์ที่ปรึกษารวมทั้งเจ้าหน้าที่ภายในหอสมุดมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรีที่เสียสละเวลาให้ความรู้ความเข้าใจ ทั้งในเรื่องการเก็บข้อมูลและคolley แนะนำวิธีการจัดการกับปัญหาต่างๆที่เกิดขึ้น นำมาสู่การทำทั่วข้อปฏิญญา妮พนธฉบับนี้ให้สำเร็จตามที่ต้องการ

## สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	ix
สารบัญตาราง	x
สารบัญรูปภาพ	xii
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1    คำสำคัญ	1
1.2    ความสำคัญของปัญหา	1
1.3    ประเภทของโครงงาน	1
1.4    วิธีการที่นำเสนอ	1
1.5    วัตถุประสงค์	2
1.6    ขอบเขตของงานวิจัย	2
1.7    เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ	2
1.8    การแยกย่อยงาน และร่างแผนการดำเนินงาน	3
1.9    ตารางการดำเนินงาน	4
1.9.1    ผลการดำเนินงานในภาคการศึกษาที่ 1	5
1.9.2    ผลการดำเนินงานในภาคการศึกษาที่ 2	5
<b>บทที่ 2 ที่มา ทฤษฎีและงานวิจัยที่เกี่ยวข้อง</b>	<b>6</b>
2.1    บทนำ	6
2.2    แนวความคิดทางทฤษฎี	6
2.2.1    การเตรียมข้อมูลรูปภาพ	6
2.2.1.1    ค่อนทัว(contour)	6

2.2.1.2 การเปลี่ยนแปลงทางสัณฐานวิทยา(Morphology Transformation)	7
2.2.2 Optical character recognition (OCR)	7
2.2.3 Natural language processing	8
2.2.3.1 Information retrieval	8
2.2.3.2 TF-IDF	9
2.2.3.3 Cosine Similarity	9
2.2.3.4 Minimum Edit Distance	10
2.2.4 RESTful Service	11
2.2.5 Word Embedding	12
2.3 ภาษาคอมพิวเตอร์และเทคโนโลยี	12
2.3.1 Open source Computer Vision (OpenCV)	12
2.3.2 Tesseract OCR	12
2.3.3 DeepCut	12
2.3.4 ReactJS	13
2.3.5 Python	13
2.3.5.1 Django	13
2.3.6 NodeJS	13
<b>บทที่ 3 การออกแบบและระบบวิจัย</b>	<b>14</b>
3.1 ภาพรวมของระบบ	14
3.2 การออกแบบการทดลอง	14
3.2.1 การแปลงหนังสือเป็นรูปภาพ	14
3.2.2 การเตรียมข้อมูลรูปภาพ	14
3.2.2.1 การข้ามหน้า	15
3.2.2.2 การหมุนรูป	16
3.2.2.3 การลบพื้นหลัง	20
3.2.3 การแปลงข้อมูลรูปภาพให้อยู่ในรูปแบบดิจิทัล	24

3.2.4	การเตรียมข้อมูลตัวหนังสือ	24
3.2.5	การสร้างแท็ก	25
3.2.5.1	การอัปเดทคำค้น TF/IDF	25
3.2.6	การค้นหา	25
3.2.7	การทำโมเดล word2vec	26
3.2.8	การจัดการหนังสือ	27
3.2.9	Login	27
3.3	System requirements	27
3.4	โครงสร้างฐานข้อมูล	29
3.4.1	Database Structure	32
3.4.2	Database Dictionary	33
3.5	UML Design	41
3.5.1	Use case diagram	41
3.5.2	Sequence diagram	42
3.5.2.1	Use case Add Document	42
3.5.2.2	Use case Manage word in document	43
3.5.2.3	Use case Verify Document to Generate Keyword	44
3.5.2.4	Use case Edit Document	45
3.5.2.5	Use case Delete Document	46
3.5.2.6	Use case View Document & Search Document	47
3.5.2.7	Use case Login	48
3.6	GUI Design	50
3.6.1	Homepage	50
3.6.2	Homepage2	51
3.6.3	Login	52
3.6.4	Insert Book(1)	53
3.6.5	Insert Book (2)	54

3.6.6	Insert Book (3)	55
3.6.7	Insert Book (4)	56
3.6.8	Insert Book (5)	57
3.6.9	Insert Book (6)	58
3.6.10	Search	59
3.6.11	Document View	60
3.6.12	Manage book	61
3.6.13	Edit Book	62
3.6.14	Upload Status Page	65
3.6.15	Evaluate Process Design	65
	<b>บทที่ 4 ผลการดำเนินงาน</b>	<b>68</b>
4.1	ผลลัพธ์ที่ได้จากการทำการเตรียมข้อมูลรูปภาพ	68
4.1.1	ผลลัพธ์ที่ได้จากการประยุกต์ใช้ OCR ในการทำ OCR ของ การทำการเตรียมข้อมูลรูปภาพ แต่ละแบบ	68
4.1.2	เปรียบเทียบประสิทธิภาพในการทำ OCR ของ การทำการเตรียมข้อมูลรูปภาพ แต่ละแบบ	68
4.1.2.1	แบบที่ 1 การใช้การข้ามหน้า, การหมุน, การลบรูปภาพ, การลบเส้น และการจัดกลุ่ม	68
4.1.2.2	แบบที่ 2 ใช้การลบพื้นหลัง	69
4.2	ผลการเปรียบเทียบข้อมูล 2 ชุด	69
4.3	ประสิทธิภาพการแก้ไขคำผิด	70
4.4	การประเมินความพึงพอใจของบรรณารักษ์ต่อการออกแบบ UX/UI	71
4.5	ผลลัพธ์จากการค้นหา	71
4.6	ผลลัพธ์ที่ได้จากการเขียนเว็บไซต์	72
4.6.1	หน้าหลัก	72
4.6.2	การเข้าสู่ระบบเว็บไซต์	72
4.6.3	การเพิ่มหนังสือเข้าสู่ระบบฐานข้อมูล	72
4.6.3.1	เพิ่มข้อมูลของหนังสือ	73
4.6.3.2	การแก้ไขและตรวจสอบคำกรองนำเข้าสู่ระบบ	75

4.6.3.3 การตรวจสอบแก้ไขแท็ก	76
4.6.4 การแสดงสถานะการเพิ่มหนังสือ	76
4.6.5 การแสดงการค้นหาหนังสือ	77
4.6.6 การแสดงข้อมูลหนังสือ	77
4.6.7 การแสดงการแก้ไขข้อมูลของหนังสือ	78
<b>บทที่ 5 สรุปผล</b>	<b>80</b>
5.1 ผลการดำเนินงาน	80
5.2 บัญหาที่พบและการแก้ไข	81
5.2.1 บัญหาน้ำเสียอ่านยาก	81
5.2.2 บัญหาระบุนไม่ตรง	81
5.2.3 บัญหาการแก้ไขคำผิด	81
5.2.4 บัญหาระยะเวลาในการเพิ่มข้อมูลหนังสือ	81
5.3 ข้อจำกัดและข้อเสนอแนะ	82
<b>หนังสืออ้างอิง</b>	<b>83</b>

## สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางการดำเนินงาน ภาคการศึกษาที่ 1/2563	4
1.2 ตารางการดำเนินงาน ภาคการศึกษาที่ 2/2563	5
2.1 Information retrieval ในลักษณะ Boolean Retrieval	8
3.1 ตารางอธิบายความหมายตาราง term_word	34
3.2 ตารางอธิบายความหมายตาราง user	34
3.3 ตารางอธิบายความหมายตาราง score	35
3.4 ตารางอธิบายความหมายตาราง pre_term_in_page	35
3.5 ตารางอธิบายความหมายตาราง page_in_document	35
3.6 ตารางอธิบายความหมายตาราง nodejs_log	36
3.7 ตารางอธิบายความหมายตาราง knex_migrations_lock	36
3.8 ตารางอธิบายความหมายตาราง knex_migrations	36
3.9 ตารางอธิบายความหมายตาราง indexing_publisher_document	37
3.10 ตารางอธิบายความหมายตาราง indexing_publisher_email_document	37
3.11 ตารางอธิบายความหมายตาราง indexing_issued_date_document	37
3.12 ตารางอธิบายความหมายตาราง indexing_creator_orgname_document	37
3.13 ตารางอธิบายความหมายตาราง indexing_creator_document	38
3.14 ตารางอธิบายความหมายตาราง dc_contributors	38
3.15 ตารางอธิบายความหมายตาราง indexing_contributor_document	38
3.16 ตารางอธิบายความหมายตาราง indexing_contributor_role_document	38
3.17 ตารางอธิบายความหมายตาราง dc_type	39
3.18 ตารางอธิบายความหมายตาราง dc_relation	39
3.19 ตารางอธิบายความหมายตาราง dc_keyword	39
3.20 ตารางอธิบายความหมายตาราง document	39
3.21 ตารางประเมินการทำ OCR	65

3.22 ตารางประเมินระบบการค้นหา	66
3.23 ตารางประเมินความพึงพอใจการออกแบบ UX/UI	66
3.24 ตารางประเมินการทำสอบเว็บไซต์	67
4.1 ตารางประเมินการทำการเตรียมข้อมูลรูปภาพแบบที่ 1	68
4.2 ตารางประเมินการทำการเตรียมข้อมูลรูปภาพแบบที่ 2	69
4.3 ตารางประเมินข้อมูลชุดที่ 1	69
4.4 ตารางประเมินข้อมูลชุดที่ 2	70
4.5 ตารางประเมินข้อมูลชุดที่ 1 ที่ไม่ผ่านการแก้ไข	70
4.6 ตารางประเมินความพึงพอใจการออกแบบ UX/UI	71
5.1 ตารางสรุปผลลัพธ์การดำเนินงาน	80

## สารบัญรูป

รูปที่		หน้า
2.1	แสดงการหาเด็ก้าโครงภายในรูป	6
2.2	แสดงการทำการขยายภาพ(dilation) เพื่อเพิ่มพื้นที่สีขาว	7
2.3	แสดงการทำกร่อนภาพ(erosion) เพื่อกร่อนพื้นที่สีขาว	7
2.4	Information retrieval ในลักษณะ Index Retrieval	9
2.5	หลักการการเช็ค edit distance [8]	10
2.6	ตัวอย่างตารางการทำ minimum edit distance [8]	10
2.7	แสดงถึงโครงสร้างของ HTTP Request [10]	11
2.8	แสดงถึงโครงสร้างของ HTTP Response [10]	12
3.1	ภาพรวมของระบบ	14
3.2	ภาพแสดงความถี่ของภาพพื้นหลังสีและภาพพื้นหลังขาวดำ	15
3.3	ภาพแสดงขั้นตอนการข้ามหน้า	15
3.4	ภาพแสดงการทำกร่อนภาพ(erosion) และการทำขยายภาพ(dilation)	16
3.5	ภาพแสดงการเปรียบเทียบการทำกร่อนภาพ(erosion) และการทำขยายภาพ(dilation)	16
3.6	ภาพแสดงเกณฑ์การวัดบรรทัดของตัวหนังสือ	17
3.7	ภาพแสดงการคัดแยกคอนทัวร์(contour) ที่ไม่ใช่ตัวหนังสือ	17
3.8	ภาพแสดงการทำ Mask ในส่วนที่ไม่ใช่ตัวหนังสือ	17
3.9	ภาพแสดงการคัดตัวหนังสือเพื่อนำไปห้องศานในการหมุน	18
3.10	ภาพแสดงการจุดของคอนทัวร์(contour) เล็กในคอนทัวร์(contour)ใหญ่	18
3.11	ภาพแสดงฟังก์ชันการลบรูปภาพออกจากหนังสือ	18
3.12	ภาพแสดงการสร้าง Mask เพื่อลบรูปภาพ	19
3.13	ภาพแสดงการสร้าง Mask โดยเว้นที่ตัวหนังสือ	19
3.14	ภาพแสดงการหาองศาในการหมุน	20
3.15	ภาพแสดงขั้นตอนในการลบพื้นหลังสี	21
3.16	รูปภาพสีก่อนถูกนำเข้ามาทำการลบพื้นหลัง	21

3.17 รูปภาพการแปลงภาพสีเป็น gray scale	22
3.18 รูปภาพที่ผ่านการทำกราฟิก โดยใช้รูปแบบสีเหลี่ยมขนาด 5x5	22
3.19 รูปภาพที่ผ่านการลบพื้นหลัง	23
3.20 รูปภาพที่ผ่านทำการ threshold แบบ THRESH_BINARY_INV	24
3.21 แสดง ER Diagram ของฐานข้อมูล	29
3.22 แสดง ER Diagram ส่วนของคีย์เวิร์ดและคะแนนความสำคัญในระบบ	29
3.23 แสดง ER Diagram ส่วนของการเก็บคำจากแต่ละหน้าที่แปลงมาจากการหนังสือ	29
3.24 แสดง ER Diagram ส่วนของประวัติของผู้ใช้งานมีการสร้างหรือแก้ไขหนังสือ	30
3.25 แสดง ER Diagram ส่วนของการเก็บข้อมูล keyword, relation, type ของหนังสือ	30
3.26 แสดง ER Diagram ส่วนของการเก็บข้อมูล Contributors ว่ามีความเกี่ยวข้องกับหนังสือในรบ้าง	30
3.27 แสดง ER Diagram ส่วนของ Creator มีความเกี่ยวข้องกับหนังสือในรบ้าง	31
3.28 แสดง ER Diagram ส่วนของ Creator Organized Name มีความเกี่ยวข้องกับหนังสือในรบ้าง	31
3.29 แสดง ER Diagram ส่วนของ Publisher มีความเกี่ยวข้องกับหนังสือในรบ้าง	31
3.30 แสดง ER Diagram ส่วนของ Publisher Email มีความเกี่ยวข้องกับหนังสือในรบ้าง	31
3.31 แสดง ER Diagram ส่วนของ Issued Date มีความเกี่ยวข้องกับหนังสือในรบ้าง	32
3.32 แสดง ER Diagram ส่วนของ Knex module ที่ใช้สำหรับ Migration ฐานข้อมูล	32
3.33 แสดง ER Diagram ส่วนของการเก็บประวัติการ HTTP Request NodeJS ไปยัง Django	32
3.34 Use case diagram	41
3.35 แสดง Scenario 1 เพิ่มหนังสือเข้าระบบ	42
3.36 แสดง Scenario 2 การจัดการคำที่ถูกเก็บได้จากหนังสือในระบบ	43
3.37 แสดง Scenario 3 ยืนหนังสือว่าพร้อมสำหรับการถูกนำไปสร้างคีย์เวิร์ด	44
3.38 แสดง Scenario 4 แก้ไขข้อมูลหนังสือ	46
3.39 แสดง Scenario 5 ลบหนังสือ	47
3.40 แสดง Scenario 6 ดูข้อมูลหนังสือ และการค้นหาหนังสือ	48
3.41 แสดง Scenario 7 ระบบล็อกอิน	49
3.42 ภาพแสดงหน้าหลักของเว็บไซต์	50

3.43	ภาพแสดงหน้าหลักของเว็บไซต์หลังจากการกดเปิดเมนู	51
3.44	ภาพแสดงหน้าเข้าสู่ระบบ	52
3.45	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นเลือกไฟล์	53
3.46	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 1	54
3.47	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 2	55
3.48	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าขั้นโหลดข้อมูลเข้าสู่ระบบ	56
3.49	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นแก้ไขคำผิด	57
3.50	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นแก้ไขและเพิ่มคำสำคัญ	58
3.51	ภาพแสดงหน้าค้นหาข้อมูล	59
3.52	ภาพแสดงหน้าดูหนังสือ	60
3.53	ภาพแสดงหน้าการจัดการหนังสือที่เพิ่มเข้าสู่ระบบ	61
3.54	ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 1	62
3.55	ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 2	63
3.56	ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 3	64
3.57	ภาพแสดงหน้าการโหลดข้อมูล	65
4.1	ภาพแสดงหน้าเว็บหลัก	72
4.2	ภาพแสดงหน้าเข้าสู่ระบบ	72
4.3	ภาพแสดงขั้นตอนการเพิ่มหนังสือขั้นตอนการเพิ่มไฟล์	73
4.4	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 1	73
4.5	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 2	74
4.6	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการเตรียมข้อมูล	74
4.7	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการแก้ไขคำผิด	75
4.8	ภาพแสดงหน้าต่างยืนยันการแก้ไข	75
4.9	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการสร้างคำสำคัญ	75
4.10	ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการแก้ไขคำสำคัญ	76

4.11 ภาพแสดงสถานะของการเพิ่มข้อมูลเข้าสู่ระบบ	76
4.12 ภาพแสดงหน้าการค้นหา	77
4.13 ภาพแสดงหน้าแสดงหนังสือ	77
4.14 ภาพแสดงข้อมูลของหนังสือ	78
4.15 ภาพแสดงหน้าการค้นหาในหน้าการจัดการหนังสือ	78
4.16 ภาพแสดงหน้าการลบหนังสือ	78
4.17 ภาพแสดงหน้าการแก้ไขข้อมูลขั้นที่ 1	79
4.18 ภาพแสดงหน้าการแก้ไขข้อมูลขั้นที่ 2	79
4.19 ภาพแสดงหน้าการแก้ไขคำสำคัญ	79

# บทที่ 1 บทนำ

## 1.1 คำสำคัญ

Natural language processing, RESTful Service, Optical character recognition, Image Processing, Information retrieval, Term Frequency-Inverse Document Frequency, Word2Vec, Word Embedded

## 1.2 ความสำคัญของปัญหา

นับตั้งแต่การก่อจังหวะสมุดมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ได้มีการเก็บรวบรวมองค์ความรู้จากประสบการณ์การทำงานของคณะอาจารย์ผู้เชี่ยวชาญในทางด้านศาสตร์ต่าง ๆ ในรูปแบบลายมือและสื่อสิ่งพิมพ์ไม่ว่าจะเป็น หนังสือ หนังสือรวมเล่มบันทึกเหตุการณ์ ในอดีตในรูปของจดหมายเหตุเพื่อส่งต่อประวัติศาสตร์ความรู้ไปยังคนรุ่นหลังโดยมีการจัดเก็บอยู่ภายใต้ห้องหมายเหตุที่มีเจ้าหน้าที่บรรณาธิการเป็นผู้ดูแล และเนื่องจากการที่ หนังสือ หนังสือยังไม่ได้มีการจัดเก็บในรูปแบบดิจิทัลทำให้เมื่อมีบุคคลภายนอกที่ต้องการข้อมูล เพื่อนำไปทำกิจกรรมต่าง ๆ ไม่ว่าจะเป็นการทำวิจัย รายงาน หรือหาข้อมูลเพื่อประกอบการประชุมก็ตามแต่ ก็จำเป็นที่จะต้องมาติดต่อเจ้าหน้าที่บรรณาธิการเป็นผู้ดูแลเพื่อที่จะให้เจ้าหน้าที่บรรณาธิการทำการค้นหาหนังสือที่มีเนื้อหาตามที่เราต้องการ ซึ่งการค้นหาข้อมูลที่ต้องการนั้นเจ้าหน้าที่จะต้องทำการค้นหาด้วยระบบมือทำให้การค้นหาข้อมูลดำเนินการไปอย่างล่าช้า นอกจากนั้นวิธีการหาข้อมูลของเจ้าหน้าที่บรรณาธิการจะเลือกตรวจสอบข้อมูลของหนังสือจากการดูสารบัญทำให้ข้อมูลที่ได้รับมาอาจจะตกหล่นจากข้อมูลเล่นอื่นได้

เพื่ออำนวยความสะดวกให้กับบรรณาธิการในการสืบค้นข้อมูลและทำให้การบริการในการสืบค้นหนังสือต่าง ๆ และให้บุคคลภายนอกสามารถทำการค้นหาข้อมูลได้ด้วยตนเองครบถ้วนทางคณิตศาสตร์จัดทำโครงการจึงได้พัฒนาระบบการจัดเก็บหนังสือและระบบการค้นหาโดยการใช้เครื่องมือในการทำ OCR เพื่อแปลงหนังสือให้อยู่ในรูปแบบของหนังสือ ดิจิทัล และหาคำสำคัญในการสร้างแท็ก ด้วยวิธี Term Frequency - Inverse Document Frequency เพื่อเพิ่มประสิทธิภาพให้กับการค้นหา

## 1.3 ประเภทของโครงงาน

นำเสนอความต้องการของผู้มีส่วนได้ส่วนเสียเฉพาะกลุ่ม

## 1.4 วิธีการที่นำเสนอ

ระบบการค้นหาหนังสือ มีขั้นตอนการทำงานดังนี้

1. นำหนังสือมาแปลงเป็นรูปภาพในรูปแบบสแกน
2. นำรูปภาพเข้าสู่ระบบโดยใช้การรับส่งข้อมูลแบบ RESTful API ในระบบประเภทของการใช้งาน
3. นำรูปภาพผ่านกระบวนการเตรียมข้อมูลรูปภาพ โดยใช้ OpenCV ในการลบส่วนอื่น ๆ ที่ไม่ใช่ข้อความออกและตัดเฉพาะข้อความเพื่อนำไปใช้ในขั้นตอน
4. นำรูปที่ผ่านการเตรียมข้อมูลรูปภาพ มาเข้าสู่ระบบ OCR เพื่อแปลงข้อมูลจากรูปภาพมาเป็นข้อความในระบบดิจิทัล
5. นำข้อมูลที่เก็บไว้มาทำการตัดแบ่งคำภาษาไทยและแก้คำผิด
6. ค้นหาคำสำคัญโดยใช้วิธี TF-IDF เพื่อนำมาใช้ในการสร้างแท็ก

7. นำข้อมูลที่ถูกแปลงเก็บและข้อมูลเกี่ยวกับแท็ก ลงในดาต้าเบส
8. ทำระบบค้นหาในรูปแบบโคไซน์ซิมิลาริตี้(Cosine Similarity)
9. ทำระบบหากำลังโดยใช้วิธี Word2Vec
10. ทำแพลตฟอร์มเว็บไซต์เพื่อเป็น User Interface ให้กับผู้ใช้งานได้ใช้งานสำหรับการใช้งานในการค้นหาข้อมูลและเพิ่มข้อมูลหนังสือลงไปในฐานข้อมูลเพิ่ม

## 1.5 วัตถุประสงค์

1. สร้างระบบแปลงข้อมูลหนังสือให้อยู่ในรูปแบบดิจิทัล
2. สร้าง web platform เพื่อทำการค้นหาหนังสือจากคำค้น และพัฒนาเครื่องมือสนับสนุนการทำงานของบรรณาธิการประจำห้องบรรณสารสนเทศ
3. สร้างระบบการค้นหาโดยการใช้วิธีการ อินฟอเมชันเรทฟวอล ซึ่งวัดความใกล้เคียงกันระหว่างคำค้นและข้อมูลในฐานข้อมูลโดยวิธีโคไซน์ซิมิลาริตี้(Cosine Similarity)
4. เพิ่มประสิทธิภาพในการเข้าถึงข้อมูลในรูปแบบดิจิทัล
5. เรียนรู้เรื่องการเตรียมข้อมูลรูปภาพ

## 1.6 ขอบเขตของงานวิจัย

1. ระบบแปลงข้อมูลจากหนังสือและหนังสือเก่า รองรับเฉพาะหนังสือที่เป็นตัวอักษรแบบพิมพ์ และรองรับไฟล์หนังสือเฉพาะ PDF เท่านั้น
2. ระบบตัดคำ Stop word ภาษาไทยโดยอ้างอิงมาจาก pythainlp และภาษาอังกฤษจาก nltk
3. ระบบค้นหาแบบโคไซน์ซิมิลาริตี้(Cosine Similarity) ในระบบ Information retrieval
4. ข้อมูลหนังสือที่นำมาใช้คือหนังสือจำพวก งานแสดงกตเวทิตาจิต หนังสือรายงานประจำปี ตั้งแต่ปี พุทธศักราช 2527 ถึง 2560 รวมประมาณ 43 เล่ม จากหอด打算หมายเหตุมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
5. ทำ platform เว็บไซต์ในรูปแบบ responsive แต่ไม่รองรับขนาดมือถือ รองรับเฉพาะคอมพิวเตอร์หรือโน้ตบุ๊ค
6. การแปลงสิ่งพิมพ์เป็นดิจิทัลใช้ Tesseract ใน การแปลงหนังสือและหนังสือเป็นรูปแบบดิจิทัล
7. การตัดคำภาษาไทยทางคณบัญชี จะใช้ freeware เช่น DeepCut มาใช้ในส่วนของการตัดคำภาษาไทย

## 1.7 เนื้อหาทางวิศวกรรมที่เป็นต้นฉบับ

- การเตรียมข้อมูลรูปภาพ สำหรับการเตรียมภาพก่อนนำไปทำ OCR

โครงการของเราทำเกี่ยวกับการทำ OCR เพื่ออ่านภาพให้กลายเป็น text แต่ถึงแม้ว่าภาพที่ได้มาจะจากการสแกนหรือการถ่ายรูป แต่ถึงอย่างนั้น OCR ที่ใช้ก็ยังคงมีข้อจำกัดในเรื่องของคุณภาพของภาพที่ใช้ ถ้าเกิดว่าภาพที่ใช้เอียง หรือมี noise จะทำให้การอ่านมีประสิทธิภาพน้อยลง นอกจากนี้การตัดภาพแยกย่อหน้าแต่ละย่อหน้าทำให้การอ่านมีความถูกต้องมากยิ่งขึ้น

- การพัฒนาเว็บไซต์สำหรับการค้นหาหนังสือในห้องสมุด

เว็บไซต์ของเราจะใช้ ReactJS, NodeJs, python ใน การพัฒนาเว็บไซต์เป็น Interface ให้กับผู้ใช้งาน สำหรับการใช้งานระบบการค้นหาหนังสือ รวมถึงการอัปโหลดหนังสือเพื่อแปลงหนังสือเข้าสู่ระบบดิจิทัลและ API ต่าง ๆ

- คัดเลือกคำสำคัญอ กมาเพื่อสร้างแท็ก

สำหรับแบ่งแยกหมวดหมู่ของหนังสือโดยใช้ หลักการของ TF-IDF ในการค้นหาคำสำคัญของหนังสือเพื่อนำมาสร้างแท็ก และใช้สำหรับการค้นหาข้อมูล

- ทำระบบค้นหาโดยใช้คำที่มีความหมายใกล้เคียง

สำหรับการค้นหาเราจะนำคะแนน TF-IDF มาใช้เป็นคะแนนเพื่อใช้ในการค้นหาแบบ cosine similarity และค้นหาคำใกล้เคียง (Query Expansion) เพื่อทำให้การค้นหาเจอกลับที่ต้องการเพิ่มมากขึ้น

## 1.8 การแยกอุปกรณ์ และร่างแผนการดำเนินงาน

- ศึกษาและค้นคว้าปัญหาของโครงการ
- เสนอหัวข้อโปรเจค
- ค้นหาข้อมูลเกี่ยวกับเทคโนโลยีที่ใช้ในโปรเจค
- ประเมินความเป็นไปได้และกำหนดขอบเขตของโปรเจค
- จัดเก็บ requirement จากกลุ่มผู้ใช้งาน
  - ติดต่อเจ้าหน้าที่ของห้องสมุด
  - เก็บข้อมูลที่ต้องการแปลงเข้าสู่ระบบดิจิทัล
- นำเสนอโครงการครั้งที่ 1
- ออกแบบ UX/UI
- การแปลงรูปภาพให้อยู่ในรูปแบบดิจิทัล
  - นำหนังสือมาแปลงเป็นรูปภาพในรูปแบบสแกน
  - ศึกษาการใช้งาน OpenCV
  - สร้างระบบการเตรียมข้อมูลรูปภาพ เพื่อทำการปรับแต่งรูปภาพและทำการปรับแต่งจนได้ระบบที่รองรับกับ Data ที่มี
  - นำรูปที่ผ่านการเตรียมข้อมูลรูปภาพ มาเข้าสู่ระบบ OCR เพื่อแปลงข้อมูลจากรูปภาพมาเป็นข้อความในระบบดิจิทัล
- นำข้อมูลที่เก็บไวมาทำการตัดแบ่งคำภาษาไทยและหาคำสำคัญโดยใช้ TF-IDF
  - ทำการตัดแบ่งคำ (Tokenization)
  - ลบ stop word ออกจากข้อมูล
- ทำระบบค้นหา
  - ทำระบบค้นหาโดยใช้หลักการ cosine similarity
  - ทำการค้นหาด้วยคำใกล้เคียงโดยใช้ Word2Vec
- จัดทำเว็บไซต์แพลตฟอร์ม
- ทดสอบระบบ

13. ปรับปรุงแก้ไข
  14. นำเสนอโครงการ

## 1.9 ตารางการดำเนินงาน

ตารางที่ 1.1 ตารางการดำเนินงาน ภาคการศึกษาที่ 1/2563

**ตารางที่ 1.2 ตารางการดำเนินงาน ภาคการศึกษาที่ 2/2563**

ที่	หัวข้อ	ตารางการดำเนินงาน ภาคการศึกษาที่ 2/2563															
		มกราคม				กุมภาพันธ์				มีนาคม				เมษายน			
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2
1	จัดทำระบบการค้นหา																
2	จัดทำเว็บไซต์																
3	ทดสอบระบบ																
4	ปรับปรุงแก้ไข																
5	นำเสนอโครงการ																

**1.9.1 ผลการดำเนินงานในภาคการศึกษาที่ 1**

- ทำระบบเตรียมข้อมูลรูปภาพ สำหรับการเตรียมรูปภาพสำหรับการแปลงข้อมูลเป็นดิจิทัล
- ทำ API ในการตัดคำและจัดการ stop word สำหรับการเตรียมการเตรียมข้อมูลตัวหนังสือ
- ทำระบบ Term Frequency-Inverse Document Frequency สำหรับการค้นหาคำสำคัญเพื่อสร้างแท็ก
- ทำส่วนของการทำการค้นหาข้อมูลเบื้องต้น

**1.9.2 ผลการดำเนินงานในภาคการศึกษาที่ 2**

- ทำระบบค้นหาให้เสร็จสิ้น
- ปรับปรุงระบบค้นหาให้ตอบโจทย์มากยิ่งขึ้น
- ทำเว็บไซต์ทั้ง front-end และ back-end

## บทที่ 2 ที่มา ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

### 2.1 บทนำ

โดยทฤษฎีที่เกี่ยวข้องกับโปรเจคนี้มีหลากหลายสาขาด้วยกันโดยจะแบ่งเป็นส่วนของการเตรียมข้อมูลรูปภาพ โดยการใช้ Open source Computer Vision (OpenCV) เพื่อนำไปใช้กับส่วนของการทำ Optical character recognition (OCR), Tesseract OCR และส่วนของการทำ Natural language processing (NLP) โดยการใช้ Team Frequency Inverse Document Frequency (TF-IDF), Minimum Edit Distance, Deep Cut ส่วนต่อไปคือสร้างระบบการค้นหาได้ใช้ cosine similarity (Cosine Similarity) และในส่วนการสร้างเว็บไซต์โดยใช้ RESTful API และส่วนสุดท้ายการทำ Word Embedding

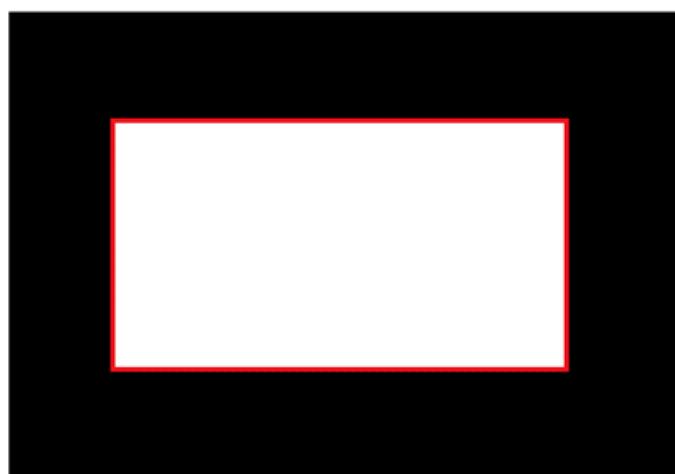
### 2.2 แนวความคิดทางทฤษฎี

#### 2.2.1 การเตรียมข้อมูลรูปภาพ

เป็นการประมวลผลรูปภาพที่แปลงภาพให้เป็นข้อมูลทางดิจิตอลเพื่อใช้สำหรับปรับคุณภาพของภาพให้ตรงตามความต้องการ อย่างการตัดสิ่งรบกวน การลบกรอบ การหมุนรูป หรือการปรับให้ภาพมีความคมชัดมากยิ่งขึ้น ในโปรเจกของเรานั้นนำมาใช้ในการปรับคุณภาพของรูปภาพเพื่อช่วยให้การทำ OCR แม่นยำมากยิ่งขึ้น

##### 2.2.1.1 คอนทัวร์(contour)

คอนทัวร์(contour) [1] คือเส้นเค้าโครงของรูปภาพ ที่วิเคราะห์ขอบเขตพื้นที่ที่มีค่าสีต่ำกว่าเนื้องอกัน หรือค่าเดียวกัน โดยใช้การเปลี่ยนให้รูปภาพอยู่ในรูปของ matrix และเช็คดูว่าค่าสีที่มีความแตกต่างอย่างชัดเจนเริ่มที่ตรงไหนและสร้างเป็นเส้นเค้าโครงขึ้นมาดังรูป 2.1 ซึ่งการทำเส้นเค้าโครงจะทำงานได้ถูกต้องเมื่อเป็นรูปภาพแบบ Binary

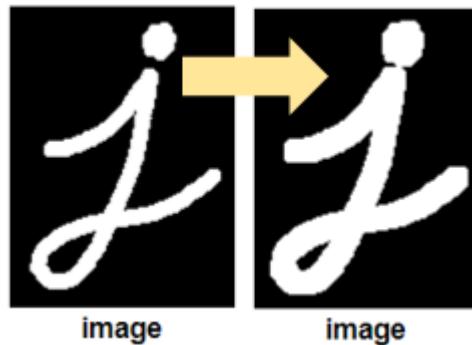


รูปที่ 2.1 แสดงการทำเค้าโครงภายในรูป

### 2.2.1.2 การเปลี่ยนแปลงทางสัณฐานวิทยา(Morphology Transformation)

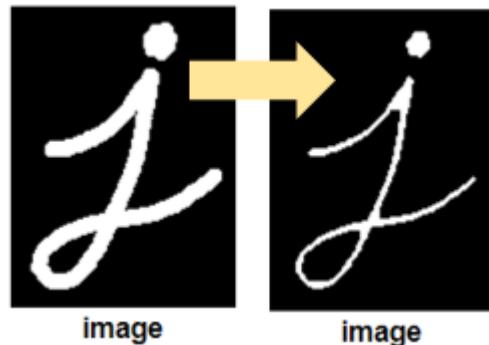
เป็นกระบวนการเตรียมข้อมูลรูปภาพ ที่จะทำการนำรูปภาพมาทำการเปลี่ยนแปลงลักษณะ รูปร่างของวัตถุภายในภาพ ปกติแล้วจะใช้ภาพที่เป็น Binary ซึ่งส่วนใหญ่จะใช้สำหรับการกำจัด noise การซ่อมแซมรูปร่างของภาพ หรือการเพิ่มขนาดให้กับวัตถุนั้นๆ โดยการทำการเปลี่ยนแปลงทางสัณฐานวิทยา(Morphology Transformation) นั้นจะมีวิธีการดำเนินการพื้นฐานอยู่ 2 วิธีคือการขยายภาพ และการกร่อนภาพ

Dilation คือการเพิ่มพื้นที่สีขาวของรูปเพิ่มพื้นที่สีขาวตามขอบพื้นที่สีขาวและจะเปลี่ยนพื้นที่สีดำให้กลายเป็นสีขาวทำให้พื้นที่สีขาวมีความหนามากขึ้นดังรูป



รูปที่ 2.2 แสดงการทำการขยายภาพ(dilation) เพื่อเพิ่มพื้นที่สีขาว

Erosion คือการกร่อนภาพ หรือก็คือจะลดพื้นที่สีขาวของภาพออกไปซึ่งวิธีการนี้ส่วนใหญ่จะใช้สำหรับการแยกสิ่งที่ของที่อยู่ติดกัน หรือลบ pepper noise ที่เป็น noise เเลิกๆได้ โดยจะใช้หลักการเดียวกับการทำขยายภาพ(dilation) เพียงแต่จะเปลี่ยนจากพื้นที่สีขาวให้กลายเป็นพื้นที่สีดำดังรูป



รูปที่ 2.3 แสดงการทำกร่อนภาพ(erosion) เพื่ogr่อนพื้นที่สีขาว

### 2.2.2 Optical character recognition (OCR)

OCR เป็นกระบวนการของการแปลงอักษรบนสื่อสิ่งพิมพ์ให้เป็นข้อความที่สามารถค้นหา เปลี่ยนแปลงและแก้ไขได้โดยที่ไม่ต้องพิมพ์ขึ้นมาใหม่ ด้วยการทำ Deep learning ใน การเรียนรู้ภาพเพื่อแปลงอักษรเป็นตัวอักษร ซึ่งในประเภทของทางผู้จัดทำต้องทำระบบเกี่ยวกับคันหาก็จะต้องคัดคำอักษรจากสื่อพิมพ์เหล่านั้น จึงจำเป็นที่จะต้องใช้ OCR ในการแปลงภาพต้นแบบอักษรให้เป็นตัวอักษรก่อนที่จะนำไปใช้งานต่อ

จากการศึกษาพบว่าการทำ OCR ภาษาไทยนั้นมีอยู่มาโดยในปัจจุบัน หนึ่งในนั้นมี T - OCR ซึ่งเป็น library ของ AI For Thai [5] และ Tesseract ของ Google [4] ที่ใช้สำหรับแปลงภาพเป็นตัวหนังสือ โดยกลุ่มของเราเลือกที่จะใช้ Tesseract ในการทำ OCR เนื่องจากไม่เสียค่าใช้จ่ายเมื่อเทียบกับการใช้ OCR ของ AI For Thai นอกจากนั้นเรื่องของการเรียกใช้งานอย่างต่อเนื่อง Tesseract สามารถทำได้ดีกว่าเนื่องจากไม่จำเป็นต้องเรียกใช้งาน AI For Thai จากภายนอก

### 2.2.3 Natural language processing

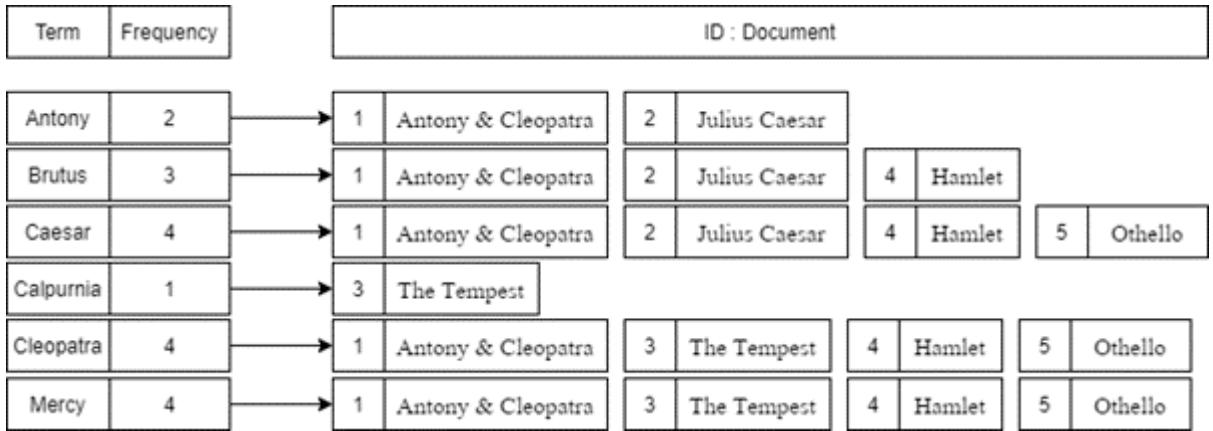
Natural language processing คือกระบวนการที่ใช้ในทางปัญญาประดิษฐ์ซึ่ง เป็นกระบวนการที่ทำการวิเคราะห์ทางด้านภาษาซึ่งเอาไปประยุกต์ทำให้ปัญญาประดิษฐ์ (AI) สามารถทำให้คอมพิวเตอร์เข้าใจภาษาและตอบกลับได้ใกล้เคียงกับมนุษย์มากขึ้น โดยในปัจจุบันนี้จะใช้มาช่วยในการหาคำสำคัญของหนังสือ และบทความต่าง ๆ เพื่อช่วยให้การค้นหาบทความมีประสิทธิภาพมากขึ้น

#### 2.2.3.1 Information retrieval

Information retrieval คือ เทคโนโลยีการเก็บข้อมูลอย่างนึงโดยจะมีทั้งหมด 2 ลักษณะ ลักษณะที่ 1 คือ Boolean Retrieval เป็นการสร้างโครงสร้างข้อมูลในรูปแบบ Matrix ที่มีค่าเพียงแค่ 0,1 โดยที่ 0 คือไม่มีคำ (Term) ในหนังสือนั้น และ 1 คือมีคำ (Term) อยู่ภายในหนังสือนั้นหรือเรียกได้วาเป็น Term-Document Incidence Matrix ดัง ตารางที่ 2.1 โดยที่ถ้าเราพิจารณาในรูปแบบแคนเราะได้ Vector ของ Term นั้นที่ปรากฏอยู่ในหนังสือ ไหนบ้าง แต่การเก็บในรูปแบบ Boolean Retrieval เมื่อมีหนังสือ เ喻อว่าจะทำให้เกิดค่า 0 ที่ไม่มีประโยชน์มากนักนั่นเองมีลักษณะที่ 2 คือโครงสร้างแบบ Inverted index เป็นการเก็บเพียง Term นั้นอยู่ภายใต้หนังสือ ไหนบ้างเพื่อจะเก็บแต่เพียงข้อมูลสำคัญเอาไว้ดัง ตารางที่ 2.2 โดย คำ (Term) จะผ่านกระบวนการเตรียมข้อมูลตัวหนังสือ ประกอบไปด้วย Tokenization (การตัดคำจากประโยชน์), Normalization (การจัดการคำย่อ), Stemming (การแปลงคำให้อยู่รูปแบบเดียวกัน), Stop words (จัดการคำที่ไม่มีความหมาย) เพื่อเป็นการจัดรูปของคำให้อยู่ในรูปแบบเดียวกันก่อนที่จะนำไปใช้งาน ซึ่งการเก็บข้อมูลแบบ Information retrieval (IR) จะทำให้การค้นหาข้อมูลภายในฐานข้อมูลได้อย่างรวดเร็วและมีประสิทธิภาพ

ตารางที่ 2.1 Information retrieval ในลักษณะ Boolean Retrieval

	Antony & Cleopatra	Julius Ceasar	The Tempest	Hamlet	Othello
Antony	1	1	0	0	0
Brutus	1	1	0	1	0
Ceasar	1	1	0	1	1
Calpurnia	0	0	1	0	0
Cleopatra	1	0	1	1	1
Mercy	1	0	1	1	1



รูปที่ 2.4 Information retrieval ในลักษณะ Index Retrieval

### 2.2.3.2 TF-IDF

เป็นเทคนิคในการคัดแยกคำตามความสำคัญผ่านการให้น้ำหนักในแต่ละคำ โดยแบ่งเป็นสองส่วนนั้นก็คือ TF (Term Frequency) เป็นการดูว่าคำนี้ หรือว่า Term นี้ปรากฏขึ้นภายใน document มากน้อยเพียงไหน และ IDF (Inverse Document Frequency) คือการหาความผกผันในความถี่ของหนังสือโดยคำนึงถึงความผกผันที่ทำให้รู้ว่าคำนี้เป็นคำที่มีความสำคัญเฉพาะภายในหนังสือนี้ แต่เนื่องจากการดูคุณแหน่ง IDF เพียงอย่างเดียวไม่สามารถบอกได้ว่า Term นั้นเป็นคำสำคัญ จึงจำเป็นต้องนำค่า TF มาคูณกับ IDF เป็นค่า TF-IDF เพื่อดูความสำคัญของ Term นั้น ในส่วนการคำนวณนี้เพื่อนำไปใช้ในการค้นหาแบบโคลาชัยมิตริตี้(Cosine Similarity) ต่อไป โดยที่ TF จะใช้เป็น Log normalization โดยคำนวนได้จากสมการ 2.1 ซึ่ง  $f_{t,d}$  คือความถี่ของคำ (Term) ที่ปรากฏขึ้นภายใน Document ส่วน IDF จะคำนวนจากสมการ 2.2 ซึ่ง  $N$  คือจำนวน Document ที่มีภายในระบบ และ  $n_t$  คือ จำนวนของ document ที่มีคำ (term) น้อยๆ เมื่อหาค่าทั้ง TF และ IDF ได้แล้วก็จะหาค่าของ TF-IDF ได้จากสมการ 2.3

$$tf = \log(1 + f_{t,d}) \quad (2.1)$$

$$idf = \log \frac{N}{n_t} \quad (2.2)$$

$$TF - IDF = tf * idf \quad (2.3)$$

### 2.2.3.3 Cosine Similarity

เป็นหน่วยวัดความคล้ายคลึงกันระหว่างข้อมูลสอง Vector โดยวัดจากมุม cosine ของจาก Vector ทั้งสองโดยคำนวนได้จากสมการ 2.4 โดยที่  $\|x\|, \|y\|$  คือ สมการของ Euclidean norm ของ Verctor  $x, y$  ดังสมการ 2.5 โดยในโปรเจคนี้เราได้นำคำน้ำหนักของ TF-IDF มาเป็นน้ำหนักในการคิดค่าโคลาชัยมิตริตี้(Cosine Similarity) โดยนำประโยชน์ที่จะค้นหามาผ่านกระบวนการเตรียมข้อมูลตัวหนังสือ ก่อนที่จะนำมาค้นหาว่า document ไหนมีค่า relevance score (คะแนนความสัมพันธ์) เพื่อนำมาเรียงค่าคะแนนสูงสุดแสดงเป็นผลลัพธ์การค้นหา

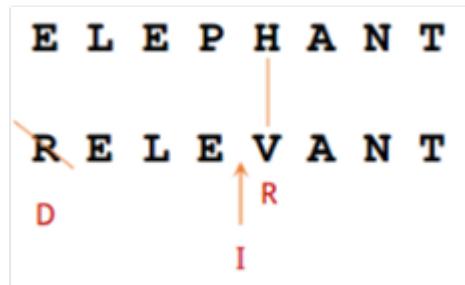
$$\sin(x, y) = \frac{x * y}{\|x\| \|y\|} \quad (2.4)$$

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \quad (2.5)$$

#### 2.2.3.4 Minimum Edit Distance

เป็นหลักการที่หารระยะห่างที่สั้นที่สุดจากคำนึงไปสู่อีกคำนึงจะมีความแตกต่างกันเท่าไหร่ซึ่งจะหลักการเช็คความห่างของคำทั้งหมดสามรูปแบบ

- รูปแบบ Insert(I) จะเป็นการเพิ่มตัวอักษรลงในคำนั้น เพื่อคำดังเดิมของเราจะเปลี่ยนแปลงเป็นคำที่เราต้องการ
- รูปแบบ Delete(D) จะเป็นการลบตัวอักษรออกในคำนั้น เพื่อคำดังเดิมของเราจะเปลี่ยนแปลงเป็นคำที่เราต้องการ
- รูปแบบ Replace(R) จะเป็นการเปลี่ยนตัวอักษรนั้นให้เป็นตัวอักษรใหม่ เพื่อคำดังเดิมของเราจะเปลี่ยนแปลงเป็นคำที่เราต้องการ



รูปที่ 2.5 หลักการการเช็ค edit distance [8]

หลังจากมีรูปแบบการวัดระยะห่างของคำดังรูปภาพที่ 2.5 และ จะต้องทำการหาคำที่สั้นที่สุดผ่านรูปแบบของตารางดังรูปภาพที่ 2.6 ซึ่งการคำนวณผ่านตารางจะเป็นการนำการกระทำก่อนหน้ามาคำนวนเรื่อยๆ จนได้รูปการเปลี่ยนเป็นคำใหม่ที่ใช้การเปลี่ยนน้อยที่สุด

	E	L	E	P	H	A	N	T
0	1	2	3	4	5	6	7	8
R	1	1	2	3	4	5	6	7
E	2	1	2	2	3	4	5	6
L	3	2	1	2	3	4	5	6
E	4	3	2	1	2	3	4	5
V	5	4	3	2	2	3	4	5
A	6	5	4	3	3	3	3	4
N	7	6	5	4	4	4	4	3
T	8	7	6	5	5	5	5	4

รูปที่ 2.6 ตัวอย่างตารางการทำ minimum edit distance [8]

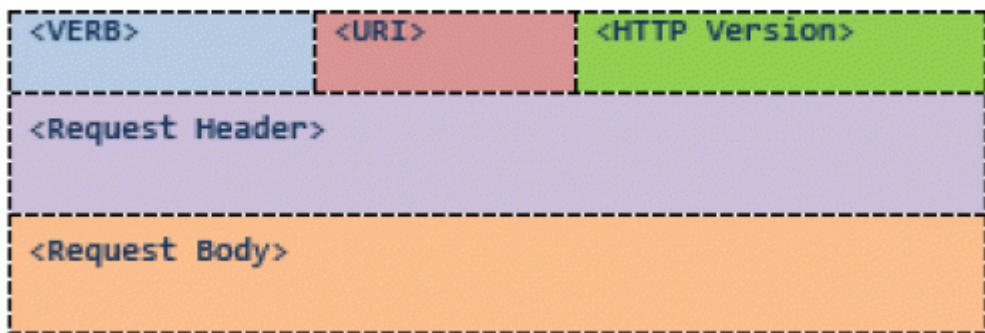
ซึ่งในโปรเจคของเราได้ตั้งหลักการ Minimum edit distance มาใช้ในการตรวจสอบหาคำที่สะกดไม่ถูกต้องโดยมีเกณฑ์ตั้งไว้ว่าถ้าเกินที่กำหนดไว้จะถือว่าคำ ๆ นั้นสะกดไม่ถูกต้องแล้วถูกแก้ให้เป็นคำที่สะกดถูกต้อง

#### 2.2.4 RESTful Service

เป็นการสร้าง web service โดยเรียกใช้ผ่านทาง HTTP Method ทั้ง 4 ประเภท GET/POST/PUT/DELETE ส่งข้อมูลอุปกรณ์เป็นรูปของ XML ทำให้ปริมาณข้อมูลที่ส่งมากน้อยกว่าการใช้ Protocol SOAP โดยโครงสร้างของ HTTP Request ดังรูปภาพที่ 2.7 ประกอบด้วย

1. VERB: แสดง method ของ HTTP
2. URI: ตำแหน่งของข้อมูลที่ต้องการ
3. HTTP Version: เวอร์ชันของ HTTP
4. Request Header: Metadata ที่เก็บข้อมูลในรูปแบบ Key-Value ของ header
5. Request Body: ส่วนเก็บข้อมูลของเนื้อหา

### HTTP Request



รูปที่ 2.7 แสดงถึงโครงสร้างของ HTTP Request [10]

HTTP Response ดังรูปภาพที่ 2.8 ประกอบด้วย

1. HTTP Version: เวอร์ชันของ HTTP
2. Response Code: รหัสผลลัพธ์ของการทำงานในระดับ HTTP เป็นเลข 3 หลัก
3. Response Header: Metadata ที่เก็บข้อมูลในรูปแบบ Key-Value ของ header
4. Response Body: ส่วนเก็บข้อมูลของเนื้อหา

## HTTP Response



รูปที่ 2.8 แสดงถึงโครงสร้างของ HTTP Response [10]

### 2.2.5 Word Embedding

เป็นวิธีการที่จะเปลี่ยนคำปกติเป็น vector ที่อยู่ในลักษณะมิติและขนาดเพื่อให้สามารถเปรียบเทียบคำต่าง ๆ ว่ามีความสัมพันธ์ใกล้เคียงกับคำไหนบ้างในระบบเพื่อที่ใช้สำหรับการทำคำที่มีความหมายใกล้เคียงกันโดยมีการทำ word embedding มากมายไม่ว่าจะเป็น Word2Vec [9] [3] ที่ถูกสร้างโดยทีมวิจัยของ Google FastText [2] เป็น word embedding อีกหนึ่งตัวที่สร้างขึ้นจากทีมวิจัยของ facebook หรือจะเป็น ELMo [7] ที่เป็นรูปแบบการ word embedding ที่ดูรูปคำโดยรอบเป็นต้น

## 2.3 ภาษาคอมพิวเตอร์และเทคโนโลยี

### 2.3.1 Open source Computer Vision (OpenCV)

เป็นซอฟต์แวร์ที่เกี่ยวกับการประมวลผลภาพที่มีการสนับสนุนการพัฒนามาจาก Intel Corporation โดยที่ตัว OpenCV นั้นเป็น Library Open Source โดยมีจุดประสงค์เพื่อให้นำไปต่อยอดการพัฒนาโปรแกรมในด้าน การรับรู้ของเห็นของคอมพิวเตอร์ (Computer Vision) ให้เข้าใจไม่ว่าจะเป็นภาพนิ่ง (Image) หรือจะเป็นภาพเคลื่อนไหว (Video) โดยภายในโปรเจคนี้ได้นำ OpenCV มาเป็นตัวทำการเตรียมข้อมูลรูปภาพ โดยที่นำรูปภาพที่ได้มาจากการสแกนหนังสือ / หนังสือ มาทำการปรับปรุงคุณภาพรูปภาพให้เหมาะสมกับการทำส่วน Optical character recognition (OCR) ให้มีความแม่นยำมากยิ่งขึ้น เช่นการลบรูปภาพ การลบสิ่งที่ลบกวน การลบกรอบตาราง การหมุนรูป

### 2.3.2 Tesseract OCR

เป็นหนึ่งใน library ที่เกี่ยวกับการทำ Optical character recognition (OCR) ที่ถูกพัฒนาโดย Google โดยเป็น Library Open Source ที่ใช้ในการทำเกี่ยวกับ Text Detection โดยสามารถเรียกใช้งานได้ผ่าน Command line หรือจะเป็นการเรียก API ภายในโปรเจกต์ ทำได้นอกจากนั้น Tesseract เวอร์ชัน 5.0.0 beta มีการใช้ Convolutional Neural Network (CNN) [6] ร่วมกันกับ Long short-term memory (LSTM) เพื่อให้การทำนายผลได้ดีขึ้นโดยเราจะนำตัว Tesseract มาทำเป็น OCR ภายในโปรเจคนี้

### 2.3.3 DeepCut

เป็น library ในภาษา python ที่สร้างมาจาก True Corporation โดยมีลักษณะเด่นที่ใช้ CNN (Convolutional Neural network) [6] มาช่วยทำให้ผลลัพธ์ที่ได้ออกมามีความแม่นยำที่ค่อนข้างสูง ซึ่งโปรเจกของเราต้องการ DeepCut เพื่อที่จะสามารถไปตัดคำจากรูปประโยค ภาษาไทยที่มีความซับซ้อน และไม่แบ่งแยกชัดเจนเหมือนภาษาอังกฤษ

### 2.3.4 ReactJS

เป็นหนึ่งใน library หรือจะเรียกว่าเป็น Framework ที่ Facebook เป็นคนสร้างขึ้นโดยทีมหน้าที่เป็นการสร้าง UI โดยมีความคิดมากจากรูปแบบ MVC [11] (Model View Controller) หรือก็คือเป็นตัวจัดการกับ Model กับ View ของตัวเว็บไซต์ โดยในโปรเจคนี้ได้เลือกใช้ ReactJS เป็น Front End สำหรับการทำ platform Web Application

### 2.3.5 Python

Python เป็นภาษาทางโปรแกรมมิ่ง ซึ่งเป็นภาษาทางคอมพิวเตอร์ระดับสูงที่ออกแบบมาให้ใกล้เคียงกับภาษามนุษย์มากที่สุดเพื่อให้สามารถเข้าใจได้่ายมากรขึ้น ซึ่งในโปรเจกมีข้อมูลที่ต้องประมวลในแต่ละครั้งมีขนาดใหญ่อ่าจะทำให้เกิดความล่าช้าในแต่ละการประมวลทางผู้จัดทำจึงเลือกใช้ python เนื่องจากการรับในส่วนของการทำ thread รวมถึงนำมาใช้ในการทำ Data preparation ทั้งการเตรียมข้อมูลรูปภาพ และการเตรียมข้อมูลต่างๆ หลังจากการทำ OCR นอกจากนี้ยังใช้ในการทำ Web server อีกด้วย

#### 2.3.5.1 Django

เป็น REST Framework ที่ใช้ภาษา python เป็นฐาน โดยในโปรเจคนี้เราจะนำมาสร้าง REST API เพื่อใช้ในการใช้ library อย่างเช่น DeepCut หรือ OCR ที่สามารถใช้ร่วมการแบ่ง multi thread ได้อย่างมีประสิทธิภาพ และยังสามารถจัดการข้อมูลใน database สำหรับโปรเจคนี้

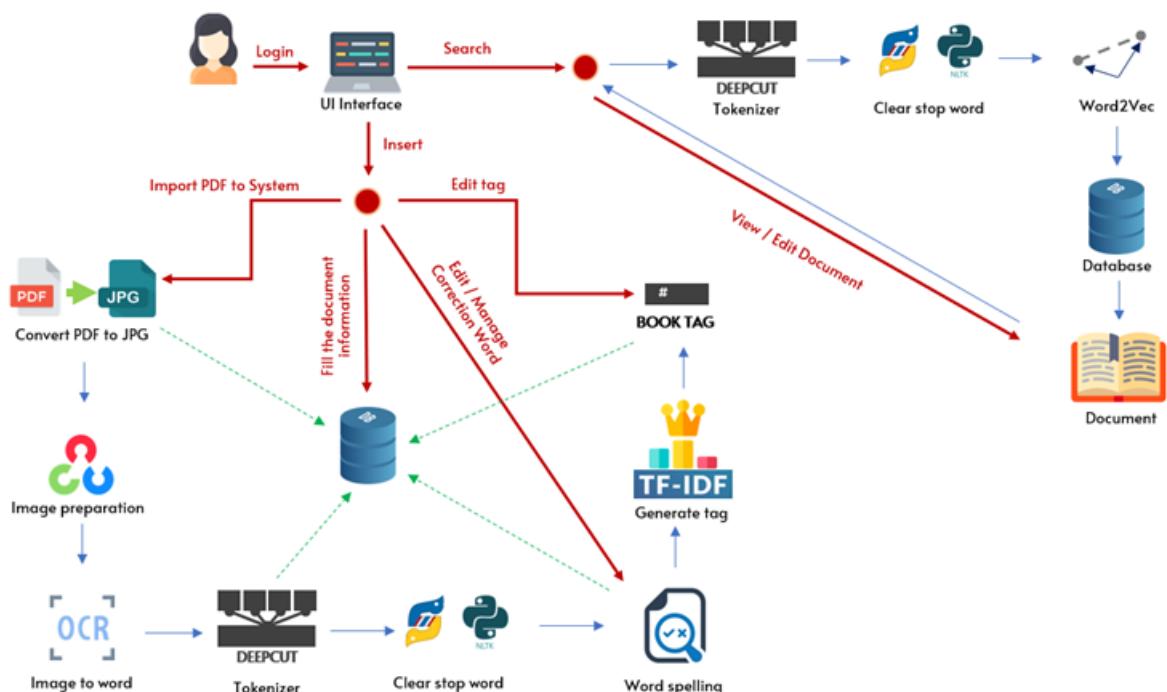
### 2.3.6 NodeJS

NodeJS เป็นสมือนแพลตฟอร์มที่ใช้ภาษา JavaScript ที่มี library สำหรับใช้จัดการกับฝั่ง Server ซึ่ง NodeJS นั้นมีความยืดหยุ่นสูงที่สำหรับการจัดการ Web server โดย library ที่นำมาใช้คือ express เป็น web server ที่เป็น RESTful API ให้

## บทที่ 3 การออกแบบและระบบประมวลผล

### 3.1 ภาพรวมของระบบ

บทนี้จะกล่าวถึงภาพรวมของระบบโดยแสดงเป็นโครงสร้างแบบรูปที่ 3.1 ซึ่งประกอบไปด้วยการออกแบบระบบฐานข้อมูล ระบบการตัดคำ ระบบการประมวลผลภาพ และการออกแบบ User Interface สำหรับการใช้งาน



รูปที่ 3.1 ภาพรวมของระบบ

### 3.2 การออกแบบการทดลอง

#### 3.2.1 การแปลงหนังสือเป็นรูปภาพ

สำหรับการแปลงหนังสือผู้ใช้จะต้องทำการอัปโหลดไฟล์ PDF ของหนังสือเข้าสู่ระบบหลังจากนั้นจะระบบจะทำการแปลงแต่ละหน้าเป็นรูปภาพ JPG เพื่อนำไปใช้ต่อในขั้นตอนต่อไปและนำไปแสดงภายใน web application

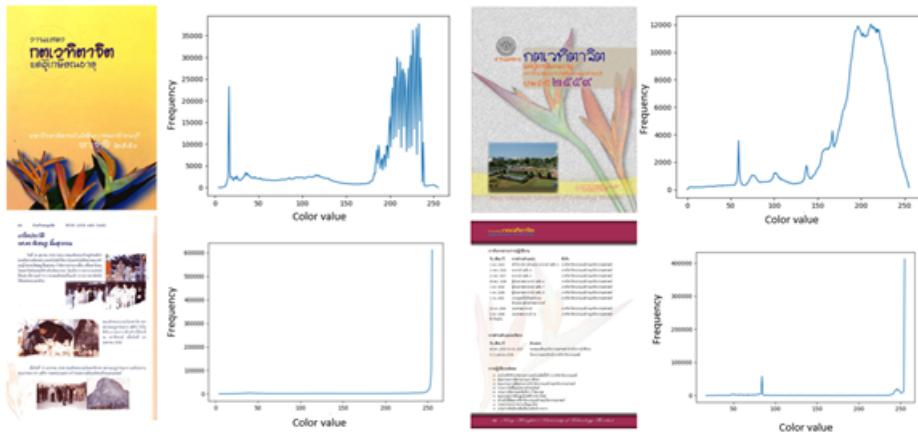
#### 3.2.2 การเตรียมข้อมูลรูปภาพ

ในส่วนของการจัดการรูปก่อนที่จะทำการ OCR ซึ่งรูปภาพนำมา OCR นั้นมาจากการสแกนทำให้ภาพส่วนใหญ่อยู่ในสภาพเดิมก็ยังคงมี noise และมีความผิดพลาดจากการสแกนเช่น ภาพเอียง หรือตัวหนังสือไม่ชัดเกิดจากการขับในระหว่างการสแกน หรือมีพื้นหลังสีที่ทำให้ OCR ไม่มีประสิทธิภาพ ดังนั้นจึงต้องมีการเตรียมข้อมูลรูปภาพก่อนที่จะไปทำ OCR ซึ่งในการเตรียมข้อมูลรูปภาพ นั้นทางคณะผู้จัดทำได้ออกแบบไว้ว่าจะทำการแยกระหว่างรูปและตัวหนังสือออกจากกัน โดยการใช้คันทัว(contour) เข้ามาช่วยในการคัดแยกรูปออกจาก

ตัวอักษร โดยดูจากพื้นที่สีเหลี่ยมที่ของคอนทัวร์(contour) กับพื้นที่คอนทัวร์(contour) ว่ามีความต่างขนาดและความแตกต่างกันมากเท่าไร หรือใช้ขนาดความกว้างและยาวมาดูว่ามีขนาดเกินเท่าไหร่จะจัดตัดให้เป็นรูปภาพ นอกจากนี้ออกแบบการหมุนโดยสร้างคอนทัวร์(contour) บรรทัดและวัดความเอียงของแต่ละบรรทัดว่าเอียงเท่าไรจากนั้นจึงหมุนกลับไปองค์กรข้าม

### 3.2.2.1 การข้ามหน้า

โดยการทำการคัดหน้าสีน้ำเงิน เราได้นำค่าความถี่ของหน้าที่มีพื้นหลังสีมาเพื่อหาความแตกต่างจะเห็นได้ว่าหน้าที่มีพื้นหลังสี หลายสี นั้นจะมีค่าความถี่กระจายอยู่หลายค่าในขณะที่ถ้าเป็นหน้าที่มีพื้นหลังสีเดียวมีค่าความถี่ในช่วงนั้นสูงทั้งรูป 3.3



รูปที่ 3.2 ภาพแสดงความถี่ของภาพพื้นหลังสีและภาพพื้นหลังขาวดำ

#### ขั้นตอนหลักการทำงานของการข้ามหน้า

1. รับไฟล์รูปภาพมาแปลงเป็น Gray Scale และทำการปรับตัวแปรที่เป็นรูปภาพแบบ Gray Scale ให้เป็นอาร์ยมิติเดียว
2. ทำการนับความถี่ของค่าสีว่าแต่ละค่าสีนั้นมีจำนวนเท่าไหร่
3. นำค่าความถี่ของค่าสีที่มีความถี่สูงสุดมาคำนวณว่ามีค่ามากกว่า 10 เปอร์เซ็นต์ของจำนวนพิกเซลทั้งหมดหรือไม่ ถ้าหากว่าให้ข้ามการทำ OCR หน้านี้ไป

```
def skipPage(image):
    #skip image or difficult bg
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    flat = gray.flatten().tolist()
    (unique, counts) = np.unique(flat, return_counts=True)
    if max(counts)/len(flat)*100 < 10:
        return True
    return False
```

รูปที่ 3.3 ภาพแสดงขั้นตอนการข้ามหน้า

### 3.2.2.2 การหมุนรูป

โดยจะมีหลักการหมุนด้วยวิธีการหาค่าองศาของแต่ละประ年之久การนำจุด 2 ที่อยู่ในแนวนอนราบเดียวกันมาทำการหา arctan เพื่อหาองศาที่ทำให้บรรทัดนั้นตรง และนำค่าองศาแต่ละบรรทัดที่อยู่ในย่อหน้าเดียวกันมาหางานาเฉลี่ยเพื่อที่จะใช้การหมุนทั้งย่อหน้าให้ตรง

#### ขั้นตอนการหมุน

- เริ่มจากนำรูปภาพมาทำเป็นสองส่วนคือการทำขยายภาพ(dilate) และกร่อนภาพ(erode) โดยที่รูปภาพที่ถูกขยายจะได้รูปที่มีการจับกลุ่มบรรทัดของย่อหน้า และรูปที่ถูกกร่อนจะได้รูปภาพที่มีการแยกบรรทัดข้อความกันอย่างชัดเจน และนำไปหาค่าองศาที่ทำให้บรรทัดนั้นตรง(contour) โดยที่รูปภาพของการจับกลุ่มบรรทัดจะได้ผลลัพธ์ดังรูปภาพที่ [3.5](#) ทางด้านซ้าย และการแยกบรรทัดจะได้ผลลัพธ์ดังรูปภาพที่ [3.5](#) ทางด้านขวา

```
#use dilate for create externalCnt | erode for
kernalDilate = cv2.getStructuringElement(cv2.MORPH_RECT,(11,5))
kernalErode = cv2.getStructuringElement(cv2.MORPH_RECT,(21,3))
dilate = cv2.dilate(imageOCR,kernalDilate,iterations=3)
erode = cv2.erode(dilate,kernalErode,iterations=3)
h,w,c = image.shape

externalCnts = findContours(dilate, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
internalCnts = findContours(erode, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
```

รูปที่ [3.4](#) ภาพแสดงการทำกร่อนภาพ(erosion) และการทำขยายภาพ(dilation)



รูปที่ [3.5](#) ภาพแสดงการเปรียบเทียบการทำกร่อนภาพ(erosion) และการทำขยายภาพ(dilation)

- ทำการแยก รูปภาพและบรรทัดข้อความ ภายในองค์นหัว(contourที่ถูกกร่อน) โดยการกำหนดค่าความสูงสูงที่สุดและต่ำที่สุดไว้ ค่าอัตราส่วนระหว่างความสูงและความกว้าง โดยค่าที่กำหนดให้เราได้มาจากการทำ Page dewarp ของ Matt Zucker [[12](#)] และนำมาตรวจสอบเงื่อนไขและแยกเป็นกลุ่มของข้อความและรูปภาพดังรูปภาพที่ [3.6](#)

```
"func rotate
TEXT_MIN_WIDTH = 15      # min reduced px width of detected text contour
TEXT_MIN_HEIGHT = 2       # min reduced px height of detected text contour
TEXT_MIN_ASPECT = 1.5     # filter out text contours below this w/h ratio
TEXT_MAX_THICKNESS = 10   # max reduced px thickness of detected text contour
TEXT_MAX_HEIGHT = 100
```

รูปที่ 3.6 ภาพแสดงเงื่อนไขการวัดบรรทัดของตัวหนังสือ

```
#find heighth and width to check that cnt is picture or text
height,width,xlow,ylow = findDistance(box,w,h)
if height < TEXT_MIN_HEIGHT or width < TEXT_MIN_WIDTH or width < TEXT_MIN_ASPECT*height or height > TEXT_MAX_HEIGHT:
    if height > TEXT_MAX_HEIGHT:
        internalNotText.append([(x+5,y+5),(x+w-5,y+5),(x+w-5,y+h-5),(x+5,y+h-5)])
        cv2.rectangle(imgNotText, (x+10, y+10), (x + w-10, y + h-10), (255,0,12),3)
    continue
internalText.append([box,xlow,ylow])
```

รูปที่ 3.7 ภาพแสดงการคัดแยกคอนทัวร์(contour) ที่ไม่ใช่ตัวหนังสือ

3. ทำการหา Mask เพื่อใช้ในการลบรูปภาพและคำนวณองศาของแต่ละบรรทัดข้อความภายในคอนทัวร์(contour)ของภาพที่ถูกขยาย(dilate)นั้น โดยการลบรูปภาพจะทำโดยการระบุว่ารูปภาพอยู่ว่าภายในคอนทัวร์(contour) ที่ถูก ไหนเพื่อจะนำมาเป็น mask สำหรับการลบรูปภาพออกตัวยิ่งการดูว่าคอนทัวร์(contour) ที่ถูกแยกเป็นรูปภาพนั้นอยู่ภายในคอนทัวร์(contour)ภาพที่ถูกขยาย(dilate) ครบทั้งสี่มุมหรือเปล่า ถ้าใช้แสดงว่าคอนทัวร์(contour)ภาพที่ถูกขยาย(dilate) คือ mask ของรูปภาพที่จะต้องถูกลบให้สร้างคอนทัวร์(contour)ภาพที่ถูกขยาย(dilate)นั้นลงไปบนรูปแต่ถ้าไม่ใช่ก็จะนำไปทางองศาของแต่บรรทัดข้อความและนำมาเฉลี่ยเป็นองศาที่ต้องหมุนสำหรับคอนทัวร์(contour)ภาพที่ถูกขยาย(dilate) นั้น

```
for exCnt in externalCnts:
    val=False
    boundaryBox = cv2.boundingRect(exCnt)
    rect = cv2.minAreaRect(exCnt)
    box = np.int0(cv2.boxPoints(rect))
    polygon = Polygon(box)
    for index, notText in enumerate(internalNotText):
        p0 = Point(notText[0])
        p1 = Point(notText[1])
        p2 = Point(notText[2])
        p3 = Point(notText[3])
        val = p1.within(polygon) and p2.within(polygon) and p3.within(polygon) and p0.within(polygon)
        if val:
            cv2.drawContours(imgNotText,[box],-1,(255,255,255),-1)
            internalNotText.pop(internalNotText.index(notText))
            break
```

รูปที่ 3.8 ภาพแสดงการทำ Mask ในส่วนที่ไม่ใช่ตัวหนังสือ

```

if not val:
    externalBox.append(box)
    externalCntBox.append(boundaryBox)
    angle = []
    avgAngle=0
if len(internalText) == 0:
    angleBox.append(avgAngle)
for indexText, textBox in enumerate(internalText):
    pi0 = Point(textBox[0][0])
    pi1 = Point(textBox[0][1])
    pi2 = Point(textBox[0][2])
    pi3 = Point(textBox[0][3])
    vali = pi1.within(polygon) and pi2.within(polygon) and pi3.within(polygon) and pi0.within(polygon)
    if vali:
        angleSingle = findAngle(textBox[1][0],textBox[2][0],textBox[1][1],textBox[2][1])
        angle.append(angleSingle)
        cv2.arrowedLine(picture,(textBox[1][0],textBox[2][0]),(textBox[1][1],textBox[2][1]),(0,255,0),2)
    if len(internalText)-1 == indexText:
        if(len(angle) != 0):
            avgAngle = findAverageAngle(angle)
            angleBox.append(avgAngle)

```

รูปที่ 3.9 ภาพแสดงการคัดตัวหนังสือเพื่อนำไปห้องศานในการหมุน



รูปที่ 3.10 ภาพแสดงการจุดของคอนทัวร์(contour) เล็กในคอนทัวร์(contour)ใหญ่

4. นำ Mask ที่เป็นรูปภาพนำมารอบออกโดยเมื่อได้ mask มา ก็จะนำไปลบออกจากรูปใหญ่ในฟังก์ชัน removePicture ในรูปที่ 3.11 ซึ่งจะนำเอาคอนทัวร์(contour)ภาพที่ถูกขยาย(dilate) ที่เป็นตัวหนังสือ (กรอบสีเขียว) มาลบออกจาก mask เดิม(กรอบสีแดง)ที่ได้สร้างไว้เพื่อกันการลบพื้นที่เป็นตัวหนังสือดัง 3.12 ด้วยการใช้ฟังก์ชัน drawContour ก่อนจะนำ mask ที่ได้มารูปออกทำให้ภาพในแต่ละหน้าหายไป เป็นผลลัพธ์ออกมาดัง 3.13

```

def removePicture(externalBox, imgNotText, image):
    for exBox in externalBox:
        cv2.drawContours(imgNotText,[exBox],-1,(0,0,0),-1)

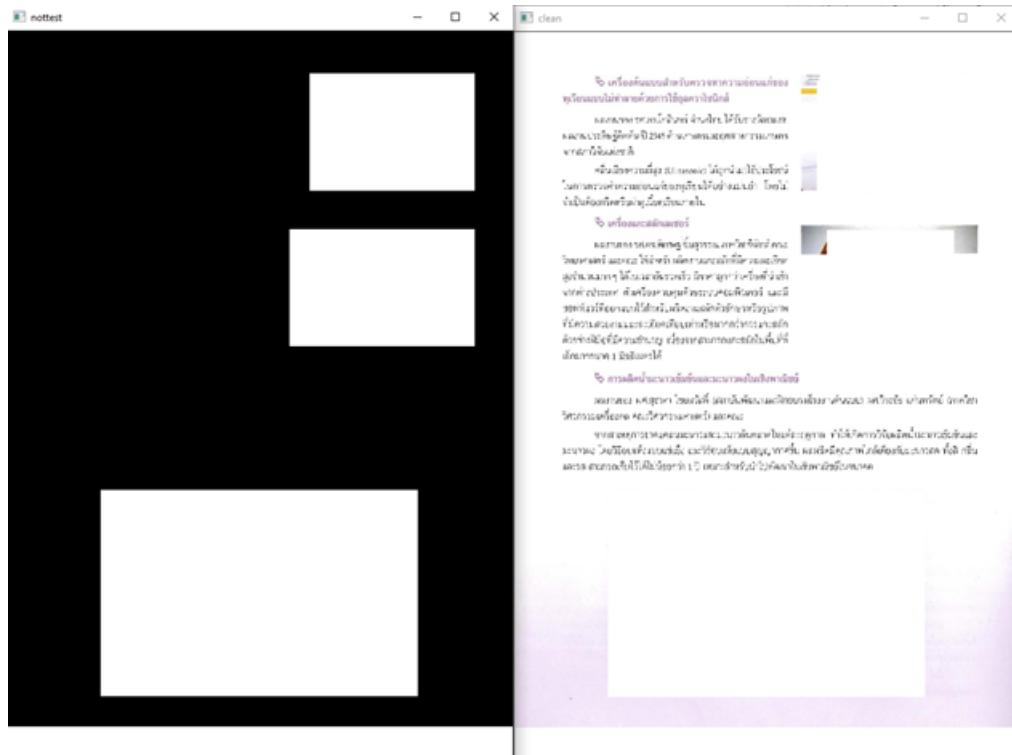
    imgNotText = cv2.cvtColor(imgNotText, cv2.COLOR_BGR2GRAY)
    deleteImage = findContours(imgNotText, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

    for delIm in deleteImage:
        cv2.drawContours(image,[delIm],-1,(255,255,255),-1)

    return image

```

รูปที่ 3.11 ภาพแสดงฟังก์ชันการลบรูปภาพออกจากหนังสือ



รูปที่ 3.12 ภาพแสดงการสร้าง Mask เพื่อลบรูปภาพ



รูปที่ 3.13 ภาพแสดงการสร้าง Mask โดยเร้นที่ตัวหนังสือ

5. การหาค่าเฉลี่ยองศาในแต่ละรูปก็จะนำค่อนทั้ง(contour) ของตัวหนังสือที่ได้มาเข้าสู่กระบวนการวัดมุมและหมุนภาพ โดยการนำจุดสีจุดของค่อนทั้ง(contour) เล็กมาเฉลี่ย องศาในการหมุน เนื่องจากค่าจุดที่ได้จากการหา minimum area rectangle นั้นอาจจะมีบางครั้งที่ค่าจุดที่ส่งมาไม่ได้เริ่มจากช้ายบน ดังนั้นจุดและเว้นที่หมายได้นั้นอาจจะทำให้องศาในแนวตั้งมาได้ จึงต้องทำการ

กรองว่าองศาที่ได้ว่าด้วยจากหรือไม่ตั้งจากถ้าเป็นองศาตั้งจากให้ทำการเอียงจากแกน 90 องศา แต่ถ้าไม่ใช่ก็เทียบจากแกนแนวอน 0 องศาหรือ 180 องศา

```
def findAngle(x1,y1,x2,y2):
    angleCal = math.degrees(math.atan2(y2-y1,x2-x1))
    return angleCal

def findAverageAngle(angle):
    for key,val in enumerate(angle):
        if val > 135.0:
            angle[key] = 180-val
        elif 135> val > 45:
            angle[key]=90-val
        elif val<-135:
            angle[key]=180+val
        elif val<-45:
            angle[key]=90+val
        else:
            angle[key]=val
    angleAvg = (sum(angle)/len(angle))
    return angleAvg
```

รูปที่ 3.14 ภาพแสดงการหาองศาในการหมุน

### 3.2.2.3 การลับพื้นหลัง

จากการที่ฟังก์ชันข้ามหน้าได้ผลลัพธ์ที่ไม่ดีพอ จึงเปลี่ยนวิธีการและขั้นตอนการเตรียมข้อมูลรูปภาพใหม่จากความรู้ที่ได้ศึกษาเพิ่มเติมขึ้น มาทำให้นำมาสู่การลับพื้นหลัง

#### ขั้นตอนการลับพื้นหลัง

1. แปลงรูปภาพสีให้กลายเป็นขาวดำจะทำให้ขั้นสีเหลือเพียงขั้นเดียวจากเดิมที่มี 3 ชั้นสีเป็น RGB เหลือเป็นค่า Gray scale ที่มีค่า 0-255 โดยที่ค่าเข้าใกล้ 0 คือสีดำและเข้าใกล้ 255 คือสีขาว
2. นำรูปภาพมาผ่านกระบวนการขยาย(dilate) โดยกำหนดเป็นสี่เหลี่ยมขนาด  $5 \times 5$  จะได้ผลลัพธ์ดังรูปภาพที่ 3.18
3. นำรูปภาพ gray scale มาหารกับค่าที่ถูกขยาย(dilate) มาโดยให้ scale อยู่ที่ 0 – 255 จะทำให้พื้นหลัง หายไปเนื่องจากจะเห็นได้ว่าถ้าส่วนไหนของรูปภาพถูกขยาย(dilate) ออกໄປจะไม่ถูกกลบเมื่อนำค่าพิกเซลล์(pixel) นั้นมาหารแต่ถ้าพื้นที่นั้นไม่ถูกขยาย(dilate) ออกໄປจะทำให้เมื่อนำค่าพิกเซลล์(pixel) มาหารจะทำให้พิกเซลล์(pixel) นั้นกลายเป็นสีขาวดังรูปภาพที่ 3.19
4. นำรูปภาพที่ไม่มีพื้นหลัง ไปทำ threshold ให้รูปเหลือเพียงสีขาวกับสีดำ โดยเราจะทำเป็น THRESH\_BINARY\_INV ที่ทำให้ตัวอักษรเป็นสีขาวและพื้นหลังเป็นสีดำเพื่อนำไปใช้ในการหา contour ต่อ

```

def removeBG(picture):
    gray = cv2.cvtColor(picture, cv2.COLOR_BGR2GRAY)
    # apply morphology
    kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5, 5))
    morph = cv2.morphologyEx(gray, cv2.MORPH_DILATE, kernel)
    # divide gray by morphology image
    division = cv2.divide(gray, morph, scale=255)
    # threshold
    return cv2.threshold(division, 0, 255, cv2.THRESH_OTSU + cv2.THRESH_BINARY_INV)[1]

```

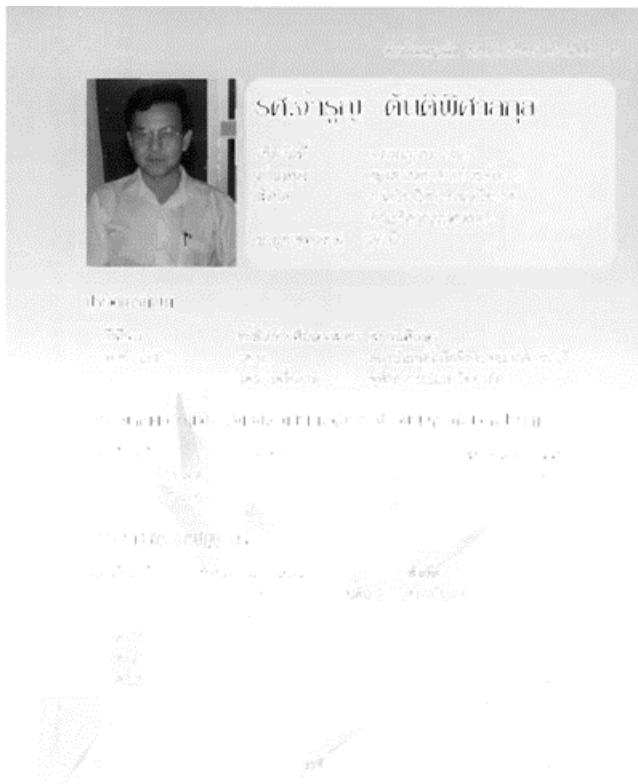
รูปที่ 3.15 ภาพแสดงขั้นตอนในการลบพื้นหลังสี



รูปที่ 3.16 รูปภาพสีก่อนถูกนำเข้ามาทำการลบพื้นหลัง



รูปที่ 3.17 รูปภาพการแปลงภาพสีเป็น gray scale



รูปที่ 3.18 รูปภาพที่ผ่านการทำกรวยๆ โดยใช้รูปแบบสี่เหลี่ยมขนาด  $5 \times 5$

ด้วยความรัก WITH LOVE AND CARE ๑



### รศ.จารุยุ ตันติพิษากุล

เดือนที่  
ตั้งแต่ง  
สังกัด  
อายุการหัวงาน

9 กันยายน 2491  
โรงพยาบาลชั้น ๑ ๖  
ภาควิชาชีวกรรมศาสตร์  
คณะวิชาชีวกรรมศาสตร์  
๓๕ ปี

#### ประวัติการทำงาน

ปี/เดือน	ชื่อ/บ้านเลขที่/ชื่อสถานศึกษา/สถานที่ทำงาน	สถานะปัจจุบัน
พ.ศ. 2516	กศ.บ.	สถาบันเทคโนโลยีพระจอมเกล้าฯ พระนครศรีอยุธยา
พ.ศ. 2520	กศ.ม. (เครื่องกล)	บริษัทลอกซ์เน็ตเวิร์คจำกัด
<b>กรณีเข้ามาที่สถาบันฯ ได้รับแต่งตั้งเป็นอาจารย์ที่ปรึกษา</b>		
วัน เดือน ปี	รายการ	สถานะปัจจุบัน
๑ มิ.ย. 2517- ๓๑ พ.ค. 2518	บริษัทลอกซ์เน็ตเวิร์คจำกัด	อาจารย์พิเศษ
๒๐ พ.ค. ๒๕๒๕- ๗ เม.ย. ๒๕๒๖	ยศน์เสนาธิการ Machinery Trade	ประทศญี่ปุ่น

#### การรับราชการ/ภารกิจ/อื่นๆ

วัน เดือน ปี	การดำรงตำแหน่ง	สังกัด
๑ มิ.ย. ๒๕๑๖	บรรจุ ตำแหน่งอาจารย์ ชั้นต้น ๒	สถาบันเทคโนโลยีพระจอมเกล้าฯ พระนครศรีอยุธยา
๑ พ.ค. ๒๕๑๘	อาจารย์ ชั้นต้น ๑	คณะวิชาชีวกรรมศาสตร์
๑ พ.ค. ๒๕๑๙	อาจารย์ ระดับ ๓	คณะวิชาชีวกรรมศาสตร์
๒๘ เม.ย. ๒๕๒๑	อาจารย์ ระดับ ๔	คณะวิชาชีวกรรมศาสตร์
๑ พ.ค. ๒๕๒๔	อาจารย์ ระดับ ๕	ภาควิชาชีวกรรมศาสตร์
๑๐ พ.ค. ๒๕๒๕	ผู้ช่วยศาสตราจารย์ ระดับ ๕	คณะวิชาชีวกรรมศาสตร์
๑ เม.ย. ๒๕๒๖	ผู้ช่วยศาสตราจารย์ ระดับ ๖	ภาควิชาชีวกรรมศาสตร์

รูปที่ 3.19 รูปภาพที่ผ่านการลบพื้นหลัง



รูปที่ 3.20 รูปภาพที่ผ่านทำการ threshold แบบ THRESH\_BINARY\_INV

### 3.2.3 การแปลงข้อมูลรูปภาพให้อยู่ในรูปแบบดิจิทัล

สำหรับการทำแปลงหนังสือเป็นข้อมูลดิจิทอลจะใช้ Tesseract OCR โดยจะใช้รูปภาพที่ผ่านกระบวนการเตรียมข้อมูลรูปภาพและประโภคที่แปลงออกมายังไฟล์เก็บไว้ใช้งานต่อไป

### 3.2.4 การเตรียมข้อมูลตัวหนังสือ

สำหรับการเตรียมข้อมูลตัวหนังสือ เราจะนำหลักการของการทำ Tokenizer มาใช้ ซึ่งหลักการนี้จะมีรูปแบบแตกต่างกันออกไปในแต่ละภาษาที่จะใช้ โดยภาษาที่เราจะใช้มีภาษาไทย และภาษาอังกฤษ อันดับแรกของกระบวนการทำคือการตัดคำจากรูปประโภคซึ่งระหว่างแต่ละภาษาจะมีรูปแบบต่างกัน เราจึงต้องนำอัลกอริทึมที่เรียกว่า Deepcut เข้ามาช่วยในการตัดแบ่งคำ หลังจากการตัดคำเราต้องจัดการกับคำที่สื่อความหมายใกล้เคียงกันแต่รูปแบบการเขียนที่แตกต่างกันอย่าง รูปแบบตัวใหญ่ตัวเล็ก คำที่อยู่ในรูปนาม กริยา กรรมแต่สื่อความหมายถึงสิ่งเดียวกัน การจัดการคำที่ไม่มีความหมายแต่เป็นคำที่รับรูปแบบการพูดเท่านั้น การแก้คำผิดสำหรับคำที่สแกนมาผิดพลาดให้กลายเป็นคำที่ถูกต้อง ซึ่งการเช็คคำในส่วนนี้จะไม่สามารถเช็คคำเฉพาะได้อย่างเช่น ชื่อเมืองที่หายไป ชื่อคน เราจึงต้องมีการเช็คคำเฉพาะอีกด้วย และส่วนสุดท้ายผู้ใช้งานจะมีสิทธิในการเช็คคำอีกรอบในขั้นตอนสุดท้ายเพื่อลดการแก้คำผิดที่ไม่สามารถแก้ได้ทั้งหมด

#### ขั้นตอนการเตรียมข้อมูลตัวหนังสือ

- นำแต่ละประโภคในหน้าบันทึกมาทำการตัดคำโดยใช้อัลกอริทึม Deepcut

2. การจัดการตัวอักษรที่เรามีต้องการใช้และถือเป็นตัวอักษรที่ผิดพลาดที่เกิดจากการทำ OCR อย่างเช่น รูปแบบตัวอักษรลatin ตัวอักษรที่เป็นรูปแบบสัญลักษณ์พิเศษ การจัดการรูปตัวอักษรตัวเล็กตัวใหญ่
3. การค้นหาคำเฉพาะโดยการดูจากบริบทของคำ กรณีภาษาอังกฤษจะเป็นคำที่มีตัวอักษรแรกเป็นตัวใหญ่ และในกรณีของการใช้ตัวอักษร “.” ของทั้งสองภาษาจะถือว่าเป็นคำเฉพาะเช่นเดียวกัน
4. นำคำที่ถูกแยกออกจากกรุ๊ปประโยคมาแทรกจากคำที่ผิดให้เป็นคำที่ถูกต้อง ซึ่งก่อนแทรกต้องระบุว่าคำที่จะแทรกเป็นภาษาชนิดใด โดยภาษาไทยจะมีวิธีการตรวจสอบโดยที่คำนั้นจะต้องมีทุกตัวอักษรเป็นภาษาไทยทั้งหมดจึงจะนับว่าเป็นภาษาไทย นอกจากนี้จากนั้นจะนับเป็นภาษาอังกฤษทั้งหมด
5. นำคำที่ถูกแก้คำแล้ว มาแก้คำเฉพาะอีกรอบนึงเนื่องจากจะมีบางคำที่ไม่สามารถแก้ได้อย่าง ชื่อคนสำคัญ ชื่อมหาวิทยาลัย โดยใช้หลักการ Minimum edit distance
6. นำผลลัพธ์ที่ได้ส่งไปยังขั้นตอนให้ผู้ใช้ตรวจสอบเพื่อเช็คคำผิดอีกรอบ
7. นำผลลัพธ์ที่ผู้ใช้งานตรวจสอบและแก้ไขมาทำการลบคำที่เป็น stop word
8. นำผลลัพธ์ที่ได้จากการทำข้อที่ 7 เข้าสู่ระบบ

### 3.2.5 การสร้างแท็ก

หลังจากการเตรียมข้อมูลตัวหนังสือ จะนำผลลัพธ์ที่ได้จะนำเข้ากระบวนการสร้างแท็ก โดยอัลกอริทึม TF-IDF ซึ่งจะถูกเก็บในรูปแบบของ Inverted index ซึ่งในกรณีหากตัวอักษรที่เราต้องการจะเป็นรูปแบบการสร้างแท็ก ของระบบที่สร้างเอง แต่จะมีในส่วนของการสร้างแท็ก ที่ผู้ใช้งานสามารถเพิ่มเข้าไปในระบบเองได้

#### ขั้นตอนการสร้างแท็ก

1. นำคำที่ได้จากการเตรียมข้อมูลตัวหนังสือ มาทำการคำนวนผ่านอัลกอริทึม TF-IDF เพื่อสร้างคะแนนขึ้นมา
2. นำแท็กผลลัพธ์ที่ได้มาให้ผู้ใช้ตรวจสอบใน 10 อันดับ และสามารถ เพิ่ม ลด แก้ไขได้ โดยที่ระบบจะทำการปรับแก้คะแนนของคำสำคัญที่ผู้ใช้งานจัดการในหนังสือ สามารถถูกค้นหาได้ง่ายกว่าคำสำคัญที่ระบบเป็นสร้างขึ้นมาเอง

#### 3.2.5.1 การอัพเดทคะแนน TF/IDF

เนื่องจากการที่มีหนังสือเพิ่มเข้ามาในระบบจะทำให้ผลลัพธ์คะแนน TF/IDF เปลี่ยนทั้งระบบจึงทำให้จำเป็นต้องมีการคำนวนใหม่เพื่อความแม่นยำในการค้นหาแต่เนื่องจากตัวระบบมีหนังสือจำนวนมากขึ้นทางผู้จัดทำจึงปรับเปลี่ยนการอัพเดทคะแนนเป็น 1 ครั้งต่อวันโดยที่คำคำนวนคะแนนในช่วงเวลาลงคืนเพื่อไม่ให้ส่งผลกระทบกับผู้ใช้งาน และในส่วนการเพิ่มข้อมูลเข้ามาใหม่คำที่อยู่ภายในหนังสือนั้นจะถูกคำนวนและอัพเดทคำใหม่ทันทีส่วนคำอื่นนั้นจะต้องรอเวลาที่กำหนดเพื่อที่จะอัพเดทคำ TF IDF และ TF/IDF

### 3.2.6 การค้นหา

ในส่วนของระบบการค้นหานั้น จะมีระบบการกรองไว้ให้ผู้ใช้สามารถเลือกได้ว่าจะค้นหาหนังสือในหัวข้อไหน ซึ่งสามารถกรองหัวข้อได้มากกว่า 1 ครั้ง ต่อการค้นหา 1 รอบ โดยระบบการค้นหาจะถูกแบ่งออกเป็นสองส่วน ในส่วนแรกจะเป็นค้นหาในรูปแบบของโคไซน์มิตริตี้(Cosine Similarity) ซึ่งจะถูกทำ Tokenizer เพื่อให้การค้นหาของผู้ใช้งานตรงกับฐานข้อมูลมากที่สุด แต่จะไม่ทำการแก้ไขคำเพื่อไม่ให้จุดประสงค์ของการค้นหาถูกเบลี่ยนไปจากเดิม หลังจากนั้นการค้นหาในระบบยังสามารถค้นหาคำใกล้เคียงที่สื่อความหมายแบบเดียวกันจากคำเดิมเพื่อที่จะได้ผลลัพธ์ที่ครอบคลุมหนังสือมากขึ้น และส่วนที่สองจะเป็นการกรองผลลัพธ์ที่ให้ผู้ใช้สามารถลดผลลัพธ์ที่ไม่เกี่ยวข้องจำนวนมากให้ลดน้อยลงเพื่อย่อต่อการค้นหา เมื่อผู้ใช้งานได้ทำการค้นหาเสร็จสิ้นระบบจะทำการส่งหนังสือที่เกี่ยวข้องกับมาให้ผู้ใช้งาน

### 3.2.7 การทำโมเดล word2vec

โดยเราได้เตรียมข้อมูลที่จะนำมาสร้างโมเดลเป็นข้อมูลหนังสือกตเวทิตาและรายงานประจำปีของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี จำนวนทั้งหมด 43 เล่ม โดยจะเป็นการนำข้อมูลที่ถูกกระบวนการ OCR และการเตรียมข้อมูลตัวหนังสือ มาตัดแบ่งคำเว้นช่องว่าง และขึ้นบรรทัดใหม่ โดยจะใช้ library genism ในสร้างโมเดลโดยมีการกำหนด window size เท่ากับ 2 และคำต้องมีกล่าวถึงมากกว่า 5 ครั้งโดยทำเป็นลักษณะ CBOW (Continuous Bag of Words)

#### ขั้นตอนการเตรียมข้อมูลที่นำมาสร้างโมเดล

1. ทำการแปลงไฟล์หนังสือ PDF ที่ต้องการเป็นไฟล์รูปภาพ
2. นำรูปภาพผ่านการเตรียมข้อมูลรูปภาพ ในการลบพื้นหลังของรูป
3. ทำการหาคันทัว(contour) ในลักษณะสี่เหลี่ยมขนาด  $5 \times 5$  เพื่อเอาไปใช้ในการตัดแบ่งประโยค
4. ทำการนำประโยคที่ถูกตัดมาทำการ OCR
5. นำแต่ละประโยคเข้ากระบวนการเตรียมข้อมูลตัวหนังสือ ในการตัดคำในรูปประโยค
6. นำคำไปจัดการตัวอักษรที่ไม่ต้องการใช้และถือเป็นตัวอักษรที่ผิดพลาดที่เกิดจากการทำ OCR
7. นำคำมาตรวจสอบแก้ไขคำผิดที่เกิดขึ้น
8. ทำการเన้นช่องว่างระหว่างคำและขั้นบรรทัดใหม่ เมื่อขั้นประโยคใหม่เพื่อที่เอามาใช้งานสำหรับการสร้างโมเดล
9. ทำการเขียนเพิ่ลงไปในไฟล์ corpus

#### ขั้นตอนการทำระบบการค้นหา

1. แบ่งแยกประเภทการค้นหาของผู้ใช้ว่ามีค้นหา และการกรองทั้งหมดกู่รูปแบบ
2. นำรูปประโยค หรือคำ จากรูปแบบการค้นหาที่ได้จากผู้ใช้ไปทำการ ตัดคำผ่าน Deepcut และหาคำที่มีความเหมือนจาก Word2vec
3. จากผลลัพธ์จะได้คำที่สามารถนำไปหาคะแนนเพื่อเข้ากระบวนการคิดแบบโคไซน์มิตริตี้(Cosine Similarity) และสามารถได้หนังสือที่เกี่ยวข้องพร้อมอันดับความเกี่ยวข้องของหนังสือนั้น ๆ
4. จากผลลัพธ์จะได้หนังสือที่เกี่ยวข้อง ดังนั้นหากผู้ใช้ต้องการกรองผลลัพธ์เหล่านี้ระบบจะต้องนำหนังสือไปตรวจสอบว่าหนังสือใดบ้างที่ตรงตามเงื่อนไข
5. นำผลลัพธ์ที่ได้ส่งคืนให้กับผู้ใช้งาน

$$sim(\vec{q}, \vec{d}) = \frac{\sum_{i=1}^{|v|} q_i d_i}{\sqrt{\sum_{i=1}^{|v|} q_i^2} \sqrt{\sum_{i=1}^{|v|} d_i^2}} \quad (3.1)$$

### 3.2.8 การจัดการหนังสือ

ในการจัดการข้อมูลหนังสือภายในระบบจะแบ่งทั้ง 3 ส่วนนั้นคือ 1. การเพิ่มหนังสือเข้าสู่ระบบ 2. การแก้ไขหนังสือภายในระบบ 3. การลบหนังสือออกจากระบบ ส่วนที่ 1. ใน การเพิ่มหนังสือเข้าสู่ระบบ ผู้ใช้งานจะต้องอัปโหลดไฟล์หนังสือในรูปแบบ PDF และกรอกรายละเอียดของหนังสือเพื่อเข้าสู่กระบวนการแปลงหนังสือเป็นรูปภาพต่อไปจะเป็นการเตรียมข้อมูลรูปภาพ ก่อนที่จะนำมาทำการแปลงภาพเป็นตัวอักษรเพื่อที่จะได้ข้อมูลจิตอลจากหนังสือที่ผู้ใช้เพิ่มเข้าสู่ระบบหลังจากนั้นจะเป็นการเตรียมข้อมูลตัวหนังสือ และให้ผู้ใช้งานได้ตรวจสอบแก้ไขหรือเพิ่มเติมก่อนจะสั่งสุดการเพิ่มหนังสือเข้าสู่ระบบ ในส่วนที่ 2 การแก้ไขหนังสือภายในระบบผู้ใช้งานสามารถค้นหาหนังสือภายในระบบเพื่อนำมาแก้ไขรายละเอียดที่ผู้ใช้งานกรอกเท่านั้นแต่สามารถแก้ไขคำที่ถูกแปลงออกมาเป็นดิจิตอลก็ครั้งได้ และส่วนสุดท้ายการลบหนังสือในระบบผู้ใช้งานสามารถลบหนังสือภายในระบบได้โดยการค้นหาหนังสือที่ต้องการและกดลบหนังสือนั้นออกจากระบบโดยเมื่อทำการลบหนังสือออก ก็จะลบคำที่มีอยู่ในหนังสือออกไปจากระบบเช่นกัน

### 3.2.9 Login

ผู้ใช้งานสามารถเข้าสู่ระบบเพื่อใช้งานฟังก์ชันต่าง ๆ ภายในระบบโดยเมื่อผู้ใช้งานทำการเข้าสู่ระบบด้วยชื่อผู้ใช้งานและรหัสผ่านแล้วจะได้รับ “token” เพื่อที่จะใช้สำหรับการยืนยันตัวในการใช้ฟังก์ชันต่าง ๆ ภายในระบบและผู้ใช้งานจะสามารถออกจากระบบได้

## 3.3 System requirements

### ผู้ใช้งาน

ใช้งานได้บนระบบ web browser

- Google Chrome เวอร์ชัน 84.0 ขึ้นไป
- Microsoft Edge เวอร์ชัน 83.0 ขึ้นไป
- Firefox เวอร์ชัน 75.0 ขึ้นไป

### ผู้ใช้ซอฟแวร์

ทางด้านฮาร์ดแวร์

- CPU: Intel or AMD processor with 64-bit โดยที่ต้องมี 4 Core ขึ้นไป
- GPU: NVIDIA 1050ti or higher
- Disk Storage: 10 GB
- RAM: 8GB or higher

ทางด้าน Software แบ่งเป็น 2 ส่วนคือ Python และ JavaScript

#### 1. Python Backend

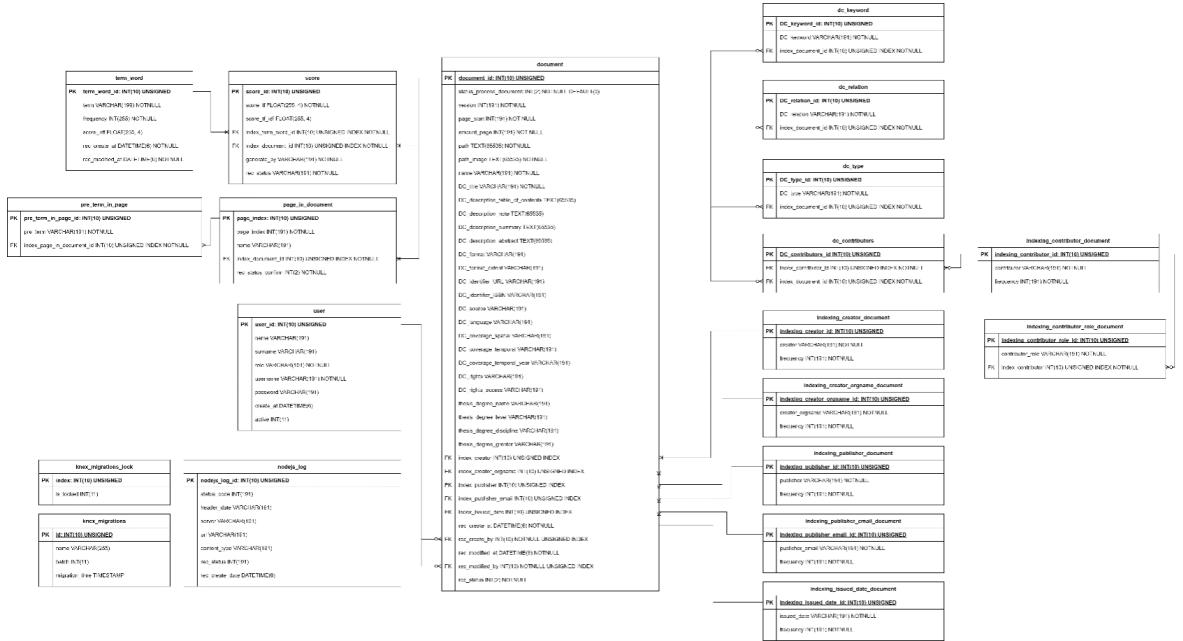
- Python เวอร์ชัน 3.7.5

- Tensorflow เวอร์ชัน 2.3.1
- DeepCut เวอร์ชัน 0.7
- Django เวอร์ชัน 3.1.3
- Djangorestframework เวอร์ชัน 3.12.2
- Django-cors-headers เวอร์ชัน 3.5.0
- Pythainlp เวอร์ชัน 2.2.5
- Pyspellchecker เวอร์ชัน 0.5.5
- nltk เวอร์ชัน 3.5.0
- mysqlclient เวอร์ชัน 2.0.1
- pillow เวอร์ชัน 8.0.1
- shapely เวอร์ชัน 1.7.1
- pytesseract เวอร์ชัน 5.0.0 beta
- opencv-python เวอร์ชัน 4.4.0.46
- pdf2image เวอร์ชัน 1.14.0
- scipy เวอร์ชัน 1.5.4

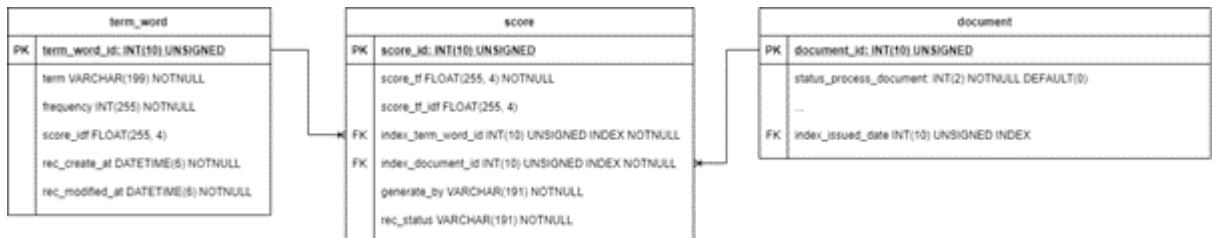
## 2. JavaScript Backend and Frontend

- nodejs เวอร์ชัน 12.16.3
- apollo-server-express เวอร์ชัน 2.19.0
- axios เวอร์ชัน 0.20.0
- cors เวอร์ชัน 2.8.5
- dotenv เวอร์ชัน 8.2.0
- express เวอร์ชัน 4.17.1
- graphql เวอร์ชัน 15.4.0
- jsonwebtoken เวอร์ชัน 8.5.1
- knex เวอร์ชัน 0.21.5
- morgan เวอร์ชัน 1.10.0
- mysql2 เวอร์ชัน 2.2.1
- password-hash เวอร์ชัน 1.2.2
- react เวอร์ชัน 16.13.1
- react-hook-form เวอร์ชัน 6.3.1
- react-router-dom เวอร์ชัน 5.2.0
- styled-components เวอร์ชัน 5.1.1
- props-types เวอร์ชัน 15.7.2

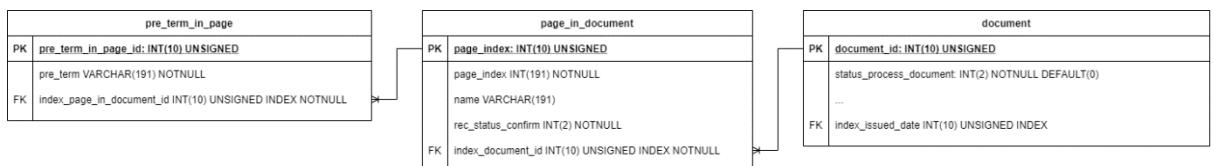
### 3.4 โครงสร้างฐานข้อมูล



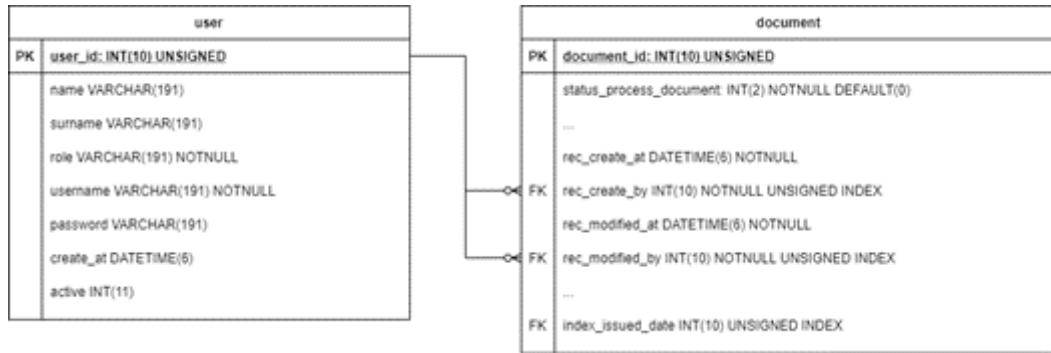
รูปที่ 3.21 แสดง ER Diagram ของฐานข้อมูล



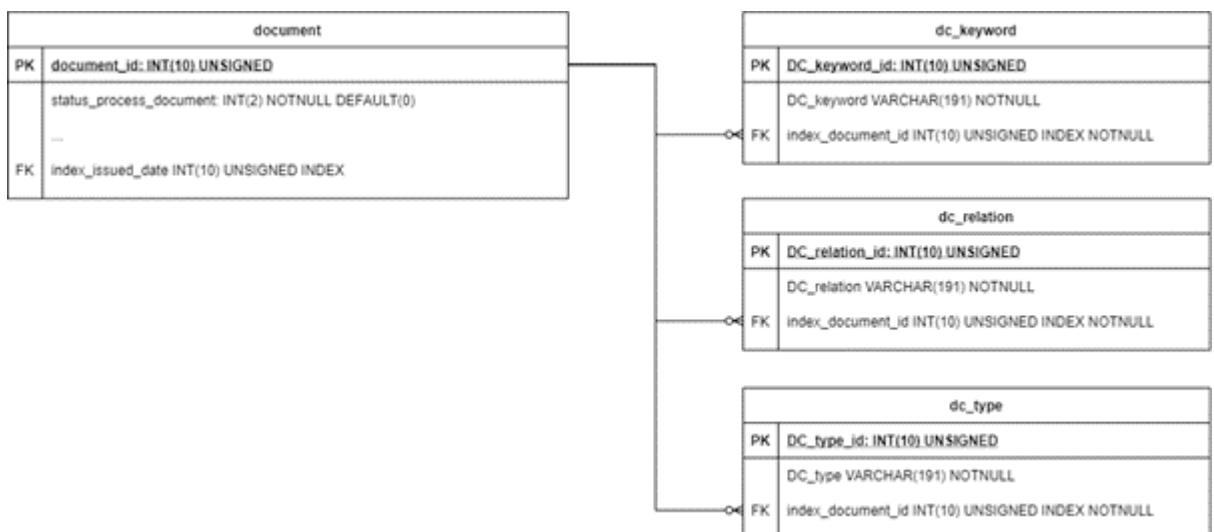
รูปที่ 3.22 แสดง ER Diagram ส่วนของคีย์เวิร์ดและคะแนนความสำคัญในระบบ



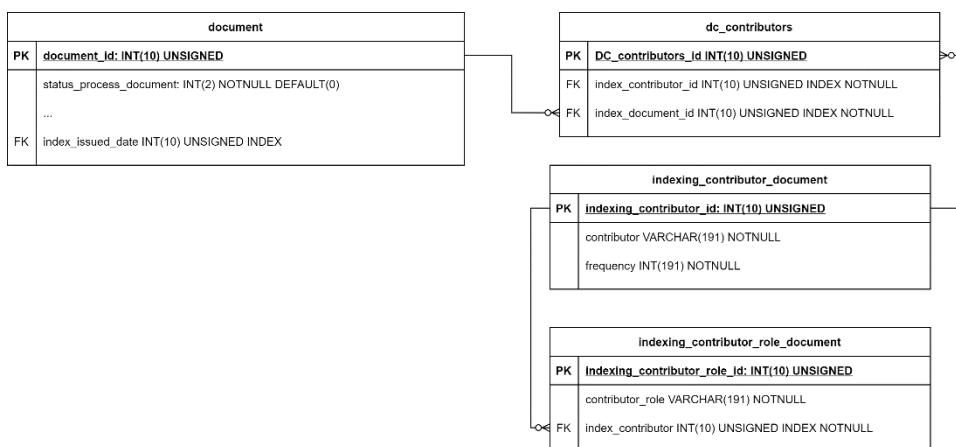
รูปที่ 3.23 แสดง ER Diagram ส่วนของการเก็บคำจากแต่ละหน้าที่แปลงมาจากหนังสือ



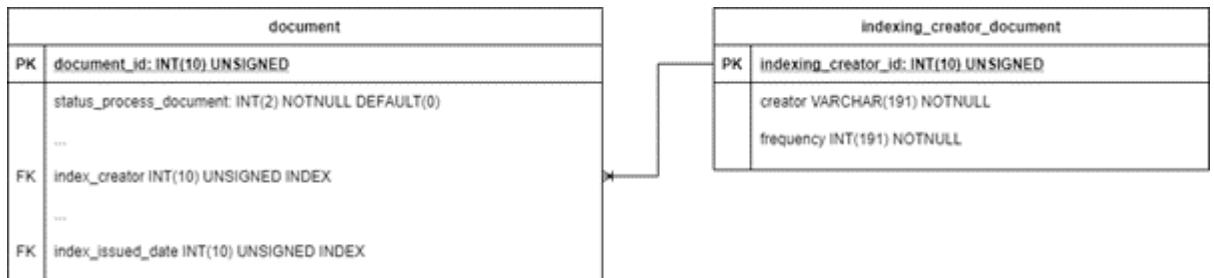
รูปที่ 3.24 แสดง ER Diagram ส่วนของประวัติของผู้ใช้งานมีการสร้างหรือแก้ไขหนังสือ



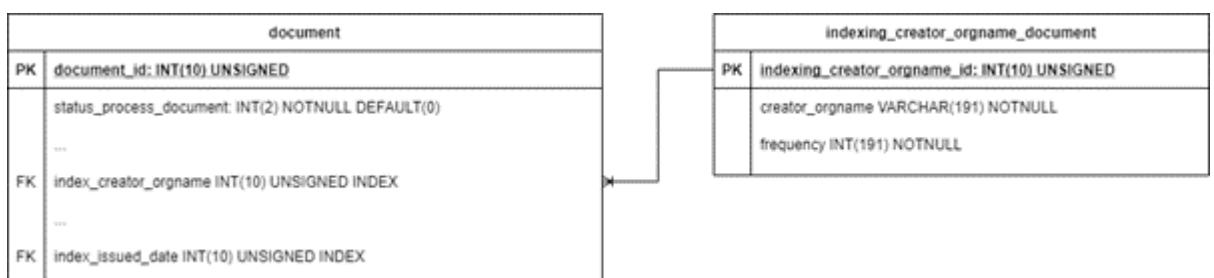
รูปที่ 3.25 แสดง ER Diagram ส่วนของการเก็บข้อมูล keyword, relation, type ของหนังสือ



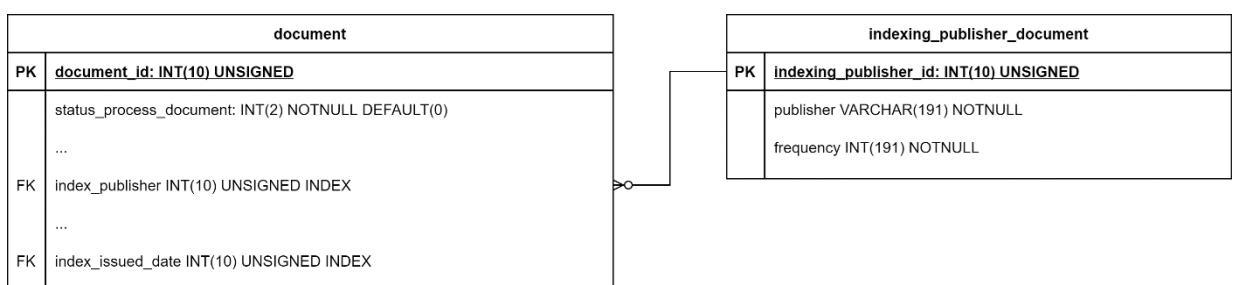
รูปที่ 3.26 แสดง ER Diagram ส่วนของการเก็บข้อมูล Contributors ว่ามีความเกี่ยวข้องกับหนังสือหรือบ้าง



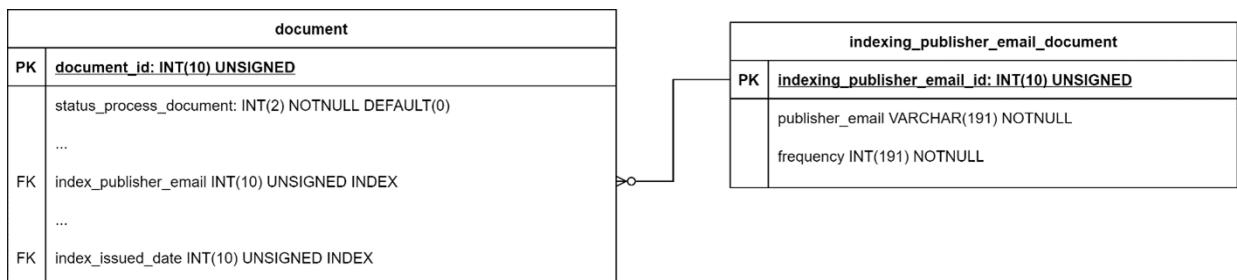
รูปที่ 3.27 แสดง ER Diagram ส่วนของ Creator มีความเกี่ยวข้องกับหนังสือในบ้าง



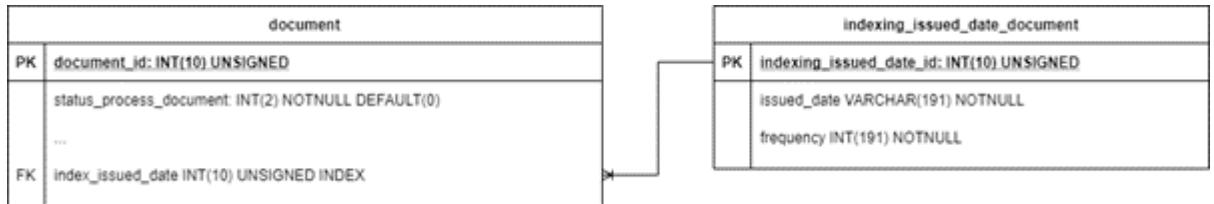
รูปที่ 3.28 แสดง ER Diagram ส่วนของ Creator Organized Name มีความเกี่ยวข้องกับหนังสือในบ้าง



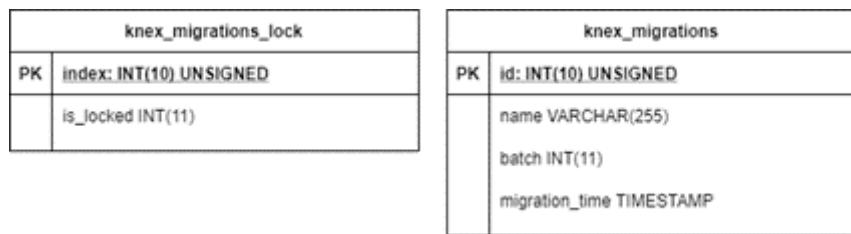
รูปที่ 3.29 แสดง ER Diagram ส่วนของ Publisher มีความเกี่ยวข้องกับหนังสือในบ้าง



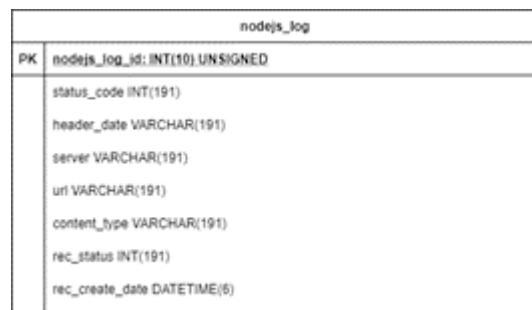
รูปที่ 3.30 แสดง ER Diagram ส่วนของ Publisher Email มีความเกี่ยวข้องกับหนังสือในบ้าง



รูปที่ 3.31 แสดง ER Diagram ส่วนของ Issued Date มีความเกี่ยวข้องกับหนังสือในหน้าบัง



รูปที่ 3.32 แสดง ER Diagram ส่วนของ Knex module ที่ใช้สำหรับ Migration ฐานข้อมูล



รูปที่ 3.33 แสดง ER Diagram ส่วนของการเก็บประวัติการ HTTP Request NodeJS ไปยัง Django

### 3.4.1 Database Structure

รูปที่ 3.21 แสดงฐานข้อมูลของทั้งระบบโดยจะมีหลัก ๆ ทั้งหมดสามส่วน ทางด้านฝั่งขวาของตาราง document จะเป็นตารางที่เก็บข้อมูลเพิ่มเติมจากตาราง document และส่วนทางด้านฝั่งซ้ายของตาราง document สำหรับการเก็บข้อมูลในด้านของการทำระบบการเก็บคำจากหนังสือที่ถูกใส่ลงมาในระบบ ระบบการแปลงคำเป็นคีย์เวิร์ดและคะแนน TF-IDF ที่นำมาใช้สำหรับการค้นหาหนังสือ ระบบจัดการฐานข้อมูลผู้ใช้งาน และการตรวจสอบความผิดพลาดที่มีโอกาสจากการสร้างคีย์เวิร์ด และส่วนสุดท้ายที่เป็นตารางที่ไม่มีการเชื่อมโยงกับตารางใด ๆ จะมีไว้สำหรับการทำระบบฐานข้อมูล และระบบตรวจสอบ HTTP Request ของทาง NodeJS

รูปที่ 3.22 จะเป็นส่วนของคีย์เวิร์ด และคะแนนเพื่อนำมาใช้สำหรับการค้นหาหนังสือของระบบนี้ โดยจะมีทั้งหมดสามตาราง document, term\_word, score ตาราง document จะเป็นตารางที่เก็บข้อมูลของหนังสือไว้ ส่วนตาราง term\_word จะเป็นการเก็บคีย์เวิร์ด และคะแนน IDF สำหรับการลดความสำคัญของคีย์เวิร์ดนั้น ๆ ไว้ซึ่งทั้งสองตารางนี้จะเป็นความสัมพันธ์แบบ one to many กับตาราง score ที่จะมีคะแนนสำหรับระบบการค้นหาเก็บเอาไว้ ที่มีความสัมพันธ์แบบนี้เนื่องจากในแต่ละคีย์เวิร์ดมีโอกาสพบได้ในหลายหนังสือ และหนังสือเองก็สามารถมีได้หลายคีย์เวิร์ด เนื่องจากแต่ละคีย์เวิร์ดที่อยู่ต่างหนังสือกันจะมีคะแนนไม่เท่ากัน

รูปที่ 3.23 จะเป็นส่วนของการเก็บคำที่แปลงมาจากหนังสือไว้โดยเริ่มที่ตาราง document จะที่สามารถบอกได้ว่าหนังสือไหน ที่จะมีความพันธ์ one to many ไปยังตาราง page\_in\_document ที่จะเป็นตารางที่บอกถึงหน้าต่าง ๆ ในหนังสือนั้น และยังมีความสัมพันธ์ one to many ต่อไปยังตาราง per\_term\_in\_page ที่จะมีคำต่าง ๆ เก็บเอาไว้ ดังนั้นจะเป็นความสัมพันธ์ที่หนังสือนั้นจะสามารถมีได้หลายหน้า แล้วแต่ละหน้าเองก็จะมีคำต่าง ๆ ที่แปลงอ กมาถูกเก็บเอาไว้

รูปที่ 3.24 จะเป็นความสัมพันธ์ของบัญชีผู้ใช้กับหนังสือ โดยจะมีตาราง user ที่จะเก็บข้อมูลของผู้ใช้งานที่มีความสัมพันธ์แบบ one to many ไปยังตาราง document ที่จะเก็บต้องเก็บข้อมูลของผู้ใช้ไว้ว่าผู้ใช้คนไหนเป็นคนสร้าง หรือแก้ไขหนังสือนี้ ซึ่งบัญชีผู้ใช้สามารถสร้าง หรือแก้ไขหนังสือได้หลายหนังสือ

รูปที่ 3.25 จะเป็นส่วนของข้อมูลของตาราง Document เมื่ອันกันแต่เนื่องจากข้อมูลมีมากกว่าหนึ่งทำให้ต้องสร้างความสัมพันธ์แบบ one to many กับตาราง dc\_keyword, dc\_relation, dc\_type ซึ่งจะเป็นข้อมูลคีย์เวิร์ด ความสัมพันธ์ และประเภทของหนังสือตามลำดับ

รูปที่ 3.26 จะเป็นการเก็บข้อมูลของ Contributor โดยจะมีตารางแยกเพื่อเก็บของสัมพันธ์ของทั้งสองด้านเนื่องจากในหนังสือสามารถมี contributor ได้หลายคน และ contributor สามารถมีหลายหนังสือเช่นกัน โดยที่ contributor จะมี role เป็นของตัวเองซึ่งสามารถมีหลาย role เช่นกันทำให้ต้องมีตาราง รองรับเพิ่ม

รูปที่ 3.27 จะเป็นส่วนของการเก็บความสัมพันธ์ระหว่าง Creator กับหนังสือ เนื่องจาก Creator สามารถมีได้หลายหนังสือทำให้ตาราง indexing\_creator\_document จะเป็นความสัมพันธ์แบบ one to many กับตาราง document

รูปที่ 3.28 จะเป็นส่วนของการเก็บความสัมพันธ์ระหว่าง Creator orgname กับหนังสือเนื่อง จากCreator orgname สามารถมีได้หลายหนังสือทำให้ตาราง indexing\_creator\_orgname\_document จะเป็นความสัมพันธ์แบบ one to many กับตาราง document

รูปที่ 3.29 จะเป็นส่วนของการเก็บความสัมพันธ์ระหว่าง Publisher กับหนังสือ เนื่องจาก Publisher สามารถมีได้หลายหนังสือทำให้ตาราง indexing\_publisher\_document จะเป็นความสัมพันธ์แบบ one to many กับตาราง document

รูปที่ 3.30 จะเป็นส่วนของการเก็บความสัมพันธ์ระหว่าง Publisher Email กับหนังสือ เนื่องจาก Publisher Email สามารถมีได้หลายหนังสือทำให้ตาราง indexing\_publisher\_email\_document จะเป็นความสัมพันธ์แบบ one to many กับตาราง document

รูปที่ 3.31 จะเป็นส่วนของการเก็บความสัมพันธ์ระหว่าง Issued Date กับหนังสือ เนื่องจาก Issued Date สามารถมีได้หลายหนังสือทำให้ตาราง indexing\_issued\_date\_document จะเป็นความสัมพันธ์แบบ one to many กับตาราง document

รูปที่ 3.32 จะเป็นสองตารางที่บันทึกการจัดการฐานข้อมูลของเครื่องมือที่ชื่อว่า Knex ที่จะทำการจัดการสร้างฐานข้อมูล ด้วยคำสั่ง Migration แล้วหลังจากทำคำสั่งเสร็จสิ้นจะเก็บบันทึกไว้

รูปที่ 3.33 จะเป็นตารางสำหรับการเก็บ HTTP Request จาก NodeJS ที่ส่งไปทางฝั่งของ Django ซึ่งจะถูกเก็บข้อมูลไว้ในตารางนี้

### 3.4.2 Database Dictionary

อธิบายถึงชื่อของคอลัมน์ ความหมายและลักษณะการเก็บข้อมูลภายในฐานข้อมูลโดยที่ตารางมีทั้งหมด 18 ตารางดังนี้

ตารางที่ 3.1 ตารางอธิบายความหมายตาราง term\_word

term_word		
ชื่อคอลัมน์	ความหมาย	ประเภท
term_word_id	id สำหรับบ่งบอกคำศัพท์	INT (10) PK Auto_Increment
term	คำศัพท์	VARCHAR (191)
frequency	จำนวนความถี่ของหนังสือที่มีคำศัพทนี้อยู่	INT (191)
score_idf	คะแนน idf ของคำศัพทนี้	FLOAT (255,4)
rec_create_at	วันเวลาของการเพิ่มคำศัพทนี้เข้าสู่ระบบ	DATETIME (6) current_timestamp
rec_modified_at	วันเวลาที่อัปเดตข้อมูลของคำศัพท์	DATETIME (6) current_timestamp

ตารางที่ 3.2 ตารางอธิบายความหมายตาราง user

user		
ชื่อคอลัมน์	ความหมาย	ประเภท
user_id	id สำหรับบ่งบอกผู้ใช้งาน	INT (10) PK Auto_Increment
name	ชื่อของผู้ใช้งาน	VARCHAR (50)
surname	นามสกุลของผู้ใช้งาน	VARCHAR (191)
role	ตำแหน่งของผู้ใช้งาน	VARCHAR (191)
username	ชื่อผู้ใช้งานสำหรับทำการ login	VARCHAR (191)
password	รหัสผ่านผู้ใช้งานสำหรับทำการ login	VARCHAR (191)
create_at	วันเวลาของผู้ใช้งานของการเพิ่มเข้าสู่ระบบ	DATETIME (6) current_timestamp
active	สถานะการรับบัญชีผู้ใช้งาน	INT (11) Default 1

ตารางที่ 3.3 ตารางอธิบายความหมายตาราง score

score		
ชื่อคอลัมน์	ความหมาย	ประเภท
score_id	id สำหรับบ่งบอกคะแนนของคำศัพท์	INT (10) PK Auto_Increment
score_tf	คะแนน tf ของคำศัพท์	FLOAT (255,4)
score_tf_idf	คะแนน tf-idf ของคำศัพท์	FLOAT (255,4)
index_term_word_id	id สำหรับบ่งบอกคำศัพท์	INT (10)
index_document_id	id สำหรับบ่งบอกหนังสือ	INT (10)
generate_by	คะแนนถูกคำนวณโดยใคร	VARCHAR (191) Default 'default'
rec_status	สถานะการใช้คะแนนนี้	INT (191) Default 1

ตารางที่ 3.4 ตารางอธิบายความหมายตาราง pre\_term\_in\_page

ชื่อคอลัมน์	ความหมาย	ประเภท
pre_term_in_page_id	id สำหรับบ่งบอกคำศัพท์ซึ่งคราวที่รอด้วยผู้ใช้งานตรวจสอบ	INT (10) PK Auto_Increment
pre_term	คำศัพท์ซึ่งคราวที่รอด้วยผู้ใช้ตรวจสอบ	VARCHAR (191)
index_page_in_document_id	id สำหรับบ่งบอกที่อยู่ของคำศัพท์ซึ่งคราวที่รอด้วยผู้ใช้งานตรวจสอบ	INT (10) FK

ตารางที่ 3.5 ตารางอธิบายความหมายตาราง page\_in\_document

page_in_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
page_in_document_id	id สำหรับบ่งบอกที่อยู่ของคำศัพท์ซึ่งคราวที่รอด้วยผู้ใช้งานตรวจสอบ	INT (10) PK Auto_Increment
page_index	หน้าของหนังสือ	INT (191)
name	ชื่อ File ของข้อมูล	VARCHAR (191)
rec_status_confirm	สถานะการยืนยันโดยผู้ใช้งาน	INT (2) PK Default 2
index_document_id	id สำหรับบ่งบอกหนังสือ	INT (10) FK

ตารางที่ 3.6 ตารางอธิบายความหมายตาราง nodejs\_log

nodejs_log		
ชื่อคอลัมน์	ความหมาย	ประเภท
nodejs_log_id	id สำหรับการจัดเก็บประวัติการทำงานผ่าน nodejs	INT (10) PK Auto_Increment
status_code	เก็บสถานะ HTTP หลังจากที่ส่งไปแล้วว่าได้สถานะใด	INT (191)
header_date	เก็บข้อมูล header ของ HTTP ที่ส่งไป	VARCHAR (191)
server	ชื่อรูปแบบของเซิฟเวอร์ที่ส่งไป	VARCHAR (191)
url	ตำแหน่งโดเมนหรือ IP ที่ส่งไป	INT (10) FK
content_type	รูปแบบเนื้อหาที่ส่งไป	VARCHAR (191)
rec_status	สถานะที่บอกว่าการส่งเกิดข้อผิดพลาดระหว่างทาง	INT (191)
rec_create_date	วันเวลาที่ทำการส่ง ณ ตอนนั้น	DATETIME (6)  current_timestamp

ตารางที่ 3.7 ตารางอธิบายความหมายตาราง knex\_migrations\_lock

knex_migrations_lock		
ชื่อคอลัมน์	ความหมาย	ประเภท
index	บ่งบอกลำดับของไฟล์ migration ของ knex	INT (10) PK Auto_Increment
is_locked	สถานะของไฟล์ migration	INT (11)

ตารางที่ 3.8 ตารางอธิบายความหมายตาราง knex\_migrations

knex_migrations		
ชื่อคอลัมน์	ความหมาย	ประเภท
id	บ่งบอกลำดับการทำงานของไฟล์ migration ของ knex	INT (10) PK Auto_Increment
name	ชื่อไฟล์ migration ที่ถูกทำงานเรียบร้อย	VARCHAR (255)
batch	ลำดับที่	INT (11)
migration_time	เวลาที่ถูกสั่งให้ทำงาน	TIMESTAMP  current_timestamp

ตารางที่ 3.9 ตารางอธิบายความหมายตาราง indexing\_publisher\_document

indexing_publisher_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_publisher_id	id สำหรับบ่งบอกสำนักพิมพ์	INT (10) PK Auto_Increment
publisher	ชื่อสำนักพิมพ์	VARCHAR (191)
frequency	จำนวนของสำนักพิมพ์ที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.10 ตารางอธิบายความหมายตาราง indexing\_publisher\_email\_document

indexing_publisher_email_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_publisher_email_id	id สำหรับบ่งบอกสำนักพิมพ์	INT (10) PK Auto_Increment
publisher_email	e-mail ของสำนักพิมพ์	VARCHAR (191)
frequency	จำนวนของสำนักพิมพ์ที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.11 ตารางอธิบายความหมายตาราง indexing\_issued\_date\_document

indexing_issued_date_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_issued_date_id	id สำหรับบ่งบอกปีที่เขียน	INT (10) PK Auto_Increment
issued_date	วันเวลาของปีที่เขียนหนังสือ	DATE
frequency	จำนวนของวันเวลาของปีที่เขียนที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.12 ตารางอธิบายความหมายตาราง indexing\_creator\_orgname\_document

indexing_creator_orgname_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_creator_orgname_id	id สำหรับบ่งบอกชื่อหน่วยงานรับผิดชอบสังกัด	INT (10) PK Auto_Increment
creator_orgname	ชื่อหน่วยงานรับผิดชอบสังกัด	VARCHAR (191)
frequency	จำนวนของหน่วยงานรับผิดชอบสังกัดที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.13 ตารางอธิบายความหมายตาราง indexing\_creator\_document

indexing_creator_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_creator_id	id สำหรับบ่งบอกชื่อผู้เขียนหนังสือ	INT (10) PK Auto_Increment
creator	ชื่อของผู้เขียนหนังสือ	VARCHAR (191)
frequency	จำนวนของผู้เขียนหนังสือที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.14 ตารางอธิบายความหมายตาราง dc\_contributors

dc_contributors		
ชื่อคอลัมน์	ความหมาย	ประเภท
dc_contributors_id	id สำหรับบ่งบอกข้อมูลความสัมพันธ์ของชื่อหน่วยข้อมูลผู้ร่วมงาน กับหนังสือ	INT (10) PK Auto_Increment
index_contributor_id	id สำหรับบ่งบอกชื่อผู้เขียนหนังสือ	INT(10) FK
index_document_id	id สำหรับบ่งบอกชื่อหน่วยข้อมูลผู้ร่วมงาน	INT(10) FK

ตารางที่ 3.15 ตารางอธิบายความหมายตาราง indexing\_contributor\_document

indexing_contributor_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_contributor_id	id สำหรับบ่งบอกชื่อหน่วยข้อมูลผู้ร่วมงาน	INT (10) PK Auto_Increment
contributor	ชื่อหน่วยข้อมูลผู้ร่วมงาน	VARCHAR (191)
frequency	จำนวนของหน่วยข้อมูลผู้ร่วมงานที่ถูกอ้างอิง	INT (191)

ตารางที่ 3.16 ตารางอธิบายความหมายตาราง indexing\_contributor\_role\_document

indexing_contributor_role_document		
ชื่อคอลัมน์	ความหมาย	ประเภท
indexing_contributor_role_id	id สำหรับบ่งบอกตำแหน่งของผู้ร่วมงาน	INT(10) Auto_Increment
contributor_role	ชื่อตำแหน่งของผู้ร่วมงาน	VARCHAR (191)
index_contributor	id สำหรับบ่งบอกชื่อหน่วยข้อมูลผู้ร่วมงาน	INT (191)

ตารางที่ 3.17 ตารางอธิบายความหมายตาราง dc\_type

dc_type		
ชื่อคอลัมน์	ความหมาย	ประเภท
DC_type_id	id สำหรับบ่งบอกประเภทของหนังสือ	INT (10) PK Auto_Increment
DC_type	ประเภทของหนังสือ	VARCHAR (191)
index_document_id	id สำหรับบ่งบอกหนังสือ	INT (10)

ตารางที่ 3.18 ตารางอธิบายความหมายตาราง dc\_relation

dc_relation		
ชื่อคอลัมน์	ความหมาย	ประเภท
DC_relation_id	id สำหรับบ่งบอกหนังสือที่เกี่ยวข้อง	INT (10) PK Auto_Increment
DC_relation	ชื่อหนังสือที่เกี่ยวข้อง	VARCHAR (191)
index_document_id	id สำหรับบ่งบอกหนังสือ	INT (10)

ตารางที่ 3.19 ตารางอธิบายความหมายตาราง dc\_keyword

dc_keyword		
ชื่อคอลัมน์	ความหมาย	ประเภท
DC_keyword_id	id สำหรับบ่งบอกคำสำคัญ	INT (10) PK Auto_Increment
DC_keyword	คำศัพท์	VARCHAR (191)
index_document_id	id สำหรับบ่งบอกหนังสือ	INT (10)

ตารางที่ 3.20: ตารางอธิบายความหมายตาราง document

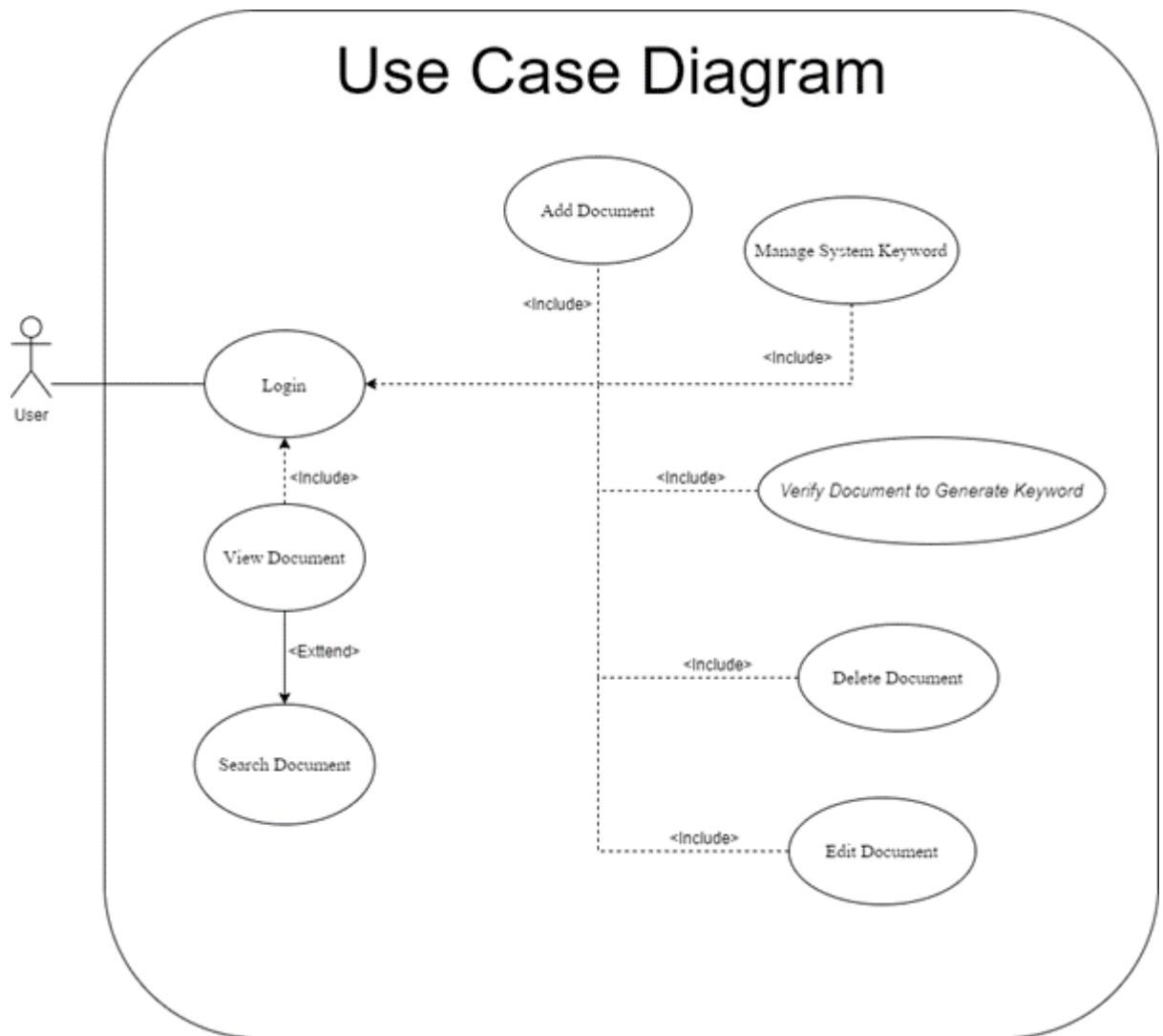
ชื่อคอลัมน์	ความหมาย	ประเภท
document_id	id สำหรับบ่งบอกหนังสือ	INT(10) Auto_Increment
status_process_document	สถานะการทำงานของหนังสือ	INT (2)
name	ชื่อไฟล์ PDF หนังสือ	VARCHAR (191)
version	ครั้งที่ตีพิมพ์	INT (191)
page_start	หน้าหนังสือที่กำหนดเป็นหน้าเริ่ม	INT (191)

amount_page	จำนวนหน้าทั้งหมดของหนังสือ	INT (191)
path	ตำแหน่งไฟล์ PDF ที่ผู้ใช้งานอัปโหลดเข้าสู่ระบบ	TEXT
path_image	ตำแหน่งไฟล์รูปภาพของหนังสือ	TEXT
name	ชื่อไฟล์ PDF หนังสือ	VARCHAR (191)
DC_title	ชื่อหนังสือ	VARCHAR (191)
DC_title_alternative	ชื่อรองของหนังสือ	VARCHAR (191)
DC_description_table_of_contents	สาระสำคัญที่มาจากการบัญ	TEXT
DC_description_note	รายละเอียดทั่วไปของหนังสือ	TEXT
DC_description_summary	สาระสำคัญของข้อมูลสารสนเทศที่ผ่านการค้นหา รวมรวม วิเคราะห์	TEXT
DC_description_abstract	ข้อมูลสรุปจากบทคัดย่อ วิทยานิพนธ์ และเนื้อหา	TEXT
DC_format	รูปแบบข้อมูลที่ถูกจัดเก็บในระบบ	VARCHAR (191)
DC_format_extent	ขนาดของไฟล์หนังสือ	VARCHAR (191)
DC_identifier_URL	แหล่งที่มาของหนังสือ	VARCHAR (191)
DC_identifier_ISBN	เลขมาตรฐานสากลของหนังสือ	VARCHAR (191)
DC_source	หน่วยข้อมูลต้นฉบับ	VARCHAR (191)
DC_language	ภาษาของหนังสือ	VARCHAR (191)
DC_coverage_spatial	สถานที่ของหนังสือที่เป็นเจ้าของ	VARCHAR (191)
DC_coverage_temporal	ช่วงเวลาในหน่วยปีของหนังสือ	VARCHAR (191)
DC_rights	ระดับการเข้าถึงของข้อมูล	VARCHAR (191)
DC_rights_access	ตำแหน่งที่มีสิทธิในการเข้าถึงข้อมูล	VARCHAR (191)
thesis_degree_name	ชื่อเต็มของปริญญา	VARCHAR (191)
thesis_degree_level	ระดับของปริญญา	VARCHAR (191)
thesis_degree_discipline	สาขาวิชา	VARCHAR (191)
thesis_degree_grantor	มหาวิทยาลัย	VARCHAR (191)
index_creator	id สำหรับบุคคลผู้เขียนหนังสือ	INT (10) FK
index_creator_orgname	id สำหรับบุคคลผู้ที่หน่วยงานรับผิดชอบสังกัด	INT (10) FK
index_publisher	id สำหรับบุคคลสำนักพิมพ์	INT (10) FK
index_publisher_email	id สำหรับบุคคลอีเมล์สำนักพิมพ์	INT (10) FK
index_issued_date	id สำหรับบุคคลที่เขียน	INT (10) FK
rec_create_at	วันเวลาของหนังสือที่ถูกนำเข้าสู่ระบบ	DATETIME (6) current_timestamp
rec_create_by	id สำหรับบุคคลผู้เขียนที่นำหนังสือเข้าสู่ระบบ	INT (10) FK

rec_modified_at	วันเวลาของหนังสือที่ถูกแก้ไขข้อมูล	DATETIME (6) current_timestamp
rec_modified_by	id สำหรับบุคคลผู้ใช้งานที่แก้ไขหนังสือในระบบ	INT (10) FK
rec_status	ค่าสถานะของหนังสือสำหรับการใช้งาน	INT (2)

### 3.5 UML Design

#### 3.5.1 Use case diagram



รูปที่ 3.34 Use case diagram

### 3.5.2 Sequence diagram

#### 3.5.2.1 Use case Add Document

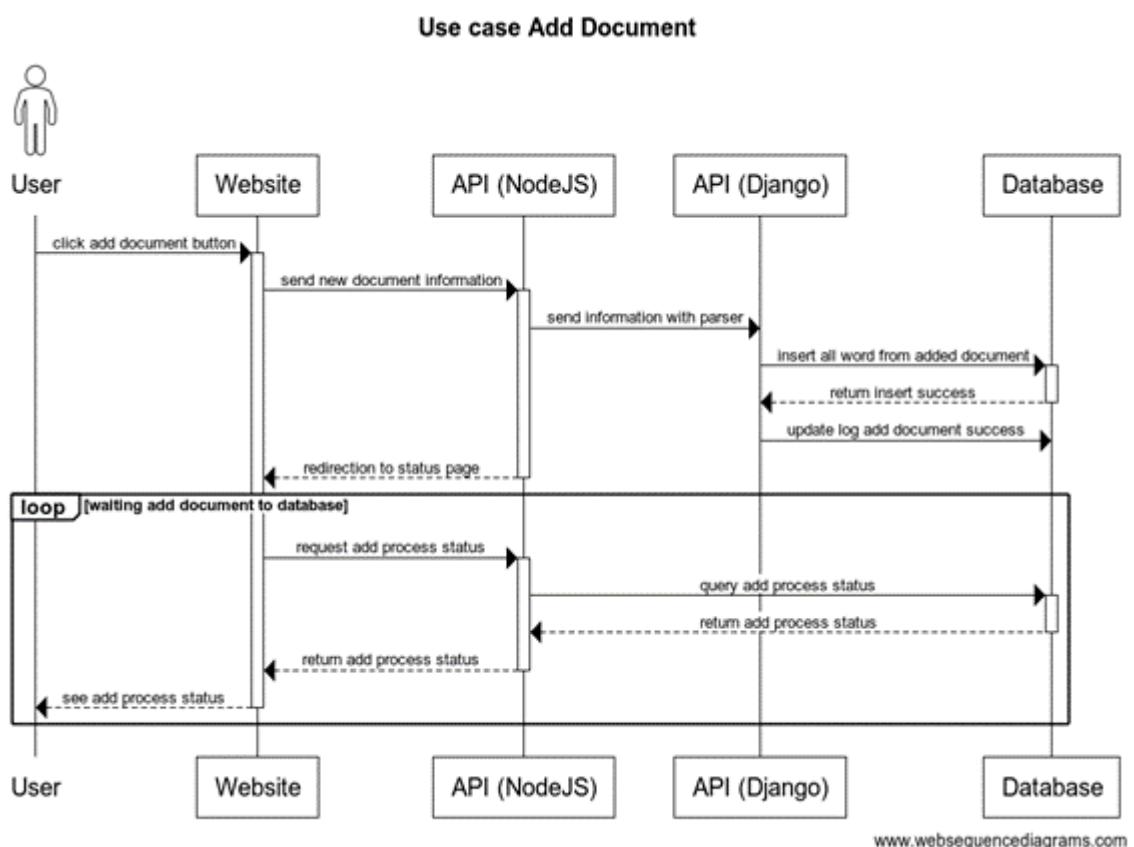
Scenario 1: เพิ่มหนังสือ/หนังสือเข้าสู่ระบบ

Goal: เพิ่มข้อมูลของหนังสือเข้าไปอยู่ในระบบ

Precondition: กดไปที่หัวข้อ INSERT BOOK ใน Web Application

Main success scenario:

1. อัปโหลดหนังสือ/หนังสือเลือกหน้าที่จะให้เริ่มต้นการแปลง
2. กรอกข้อมูลรายละเอียดที่ต้องการลงในระบบ
3. แสดงสถานะของการเพิ่มข้อมูล
4. เพิ่มหนังสือ/หนังสือเข้าสู่ระบบ



รูปที่ 3.35 แสดง Scenario 1 เพิ่มหนังสือเข้าระบบ

### 3.5.2.2 Use case Manage word in document

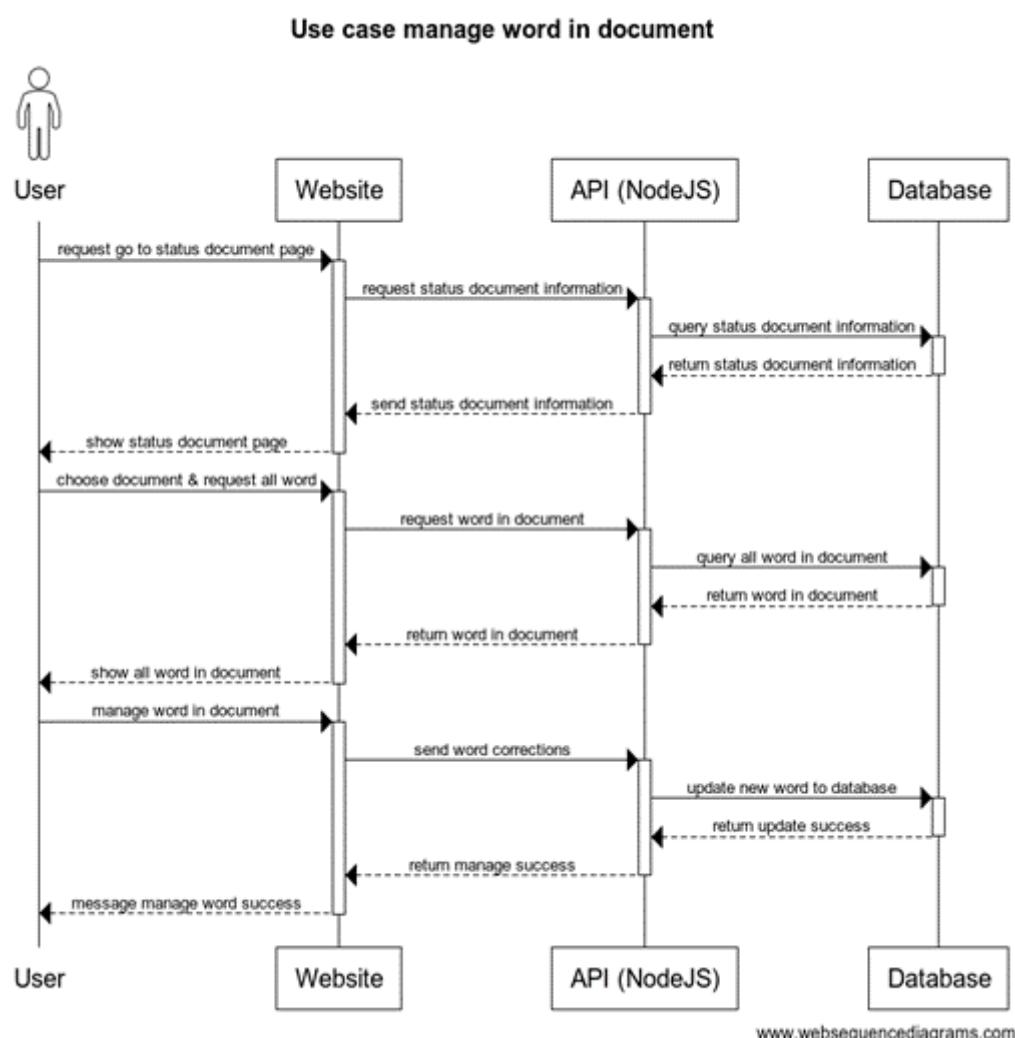
Scenario 2: การตรวจสอบและแก้ไขคำที่ถูกเก็บออกจากหนังสือในระบบ

Goal: ผู้ใช้งานเห็นคำที่จะถูกการแปลงเป็นดิจิตอลแล้วสามารถจัดการคำเหล่านี้ได้

Precondition: อุปกรณ์ภายในขั้นตอนการเพิ่มหนังสือ/หนังสือลงในระบบ

Main success scenario:

1. ผู้ใช้เข้าไปยังหน้าดูสถานะการเพิ่มหนังสือ
2. ผู้ใช้เลือกหนังสือที่อยู่ในสถานะตรวจสอบคำ
3. ระบบแสดงคำทั้งหมดที่ถูกแปลงมาได้จากหนังสือแต่ละหน้า
4. ผู้ใช้ตรวจสอบ แก้ไขคำที่แสดงขึ้นมา
5. ยืนยันขั้นตอนการตรวจสอบและแก้ไขคำ



รูปที่ 3.36 แสดง Scenario 2 การจัดการคำที่ถูกเก็บออกจากหนังสือในระบบ

### 3.5.2.3 Use case Verify Document to Generate Keyword

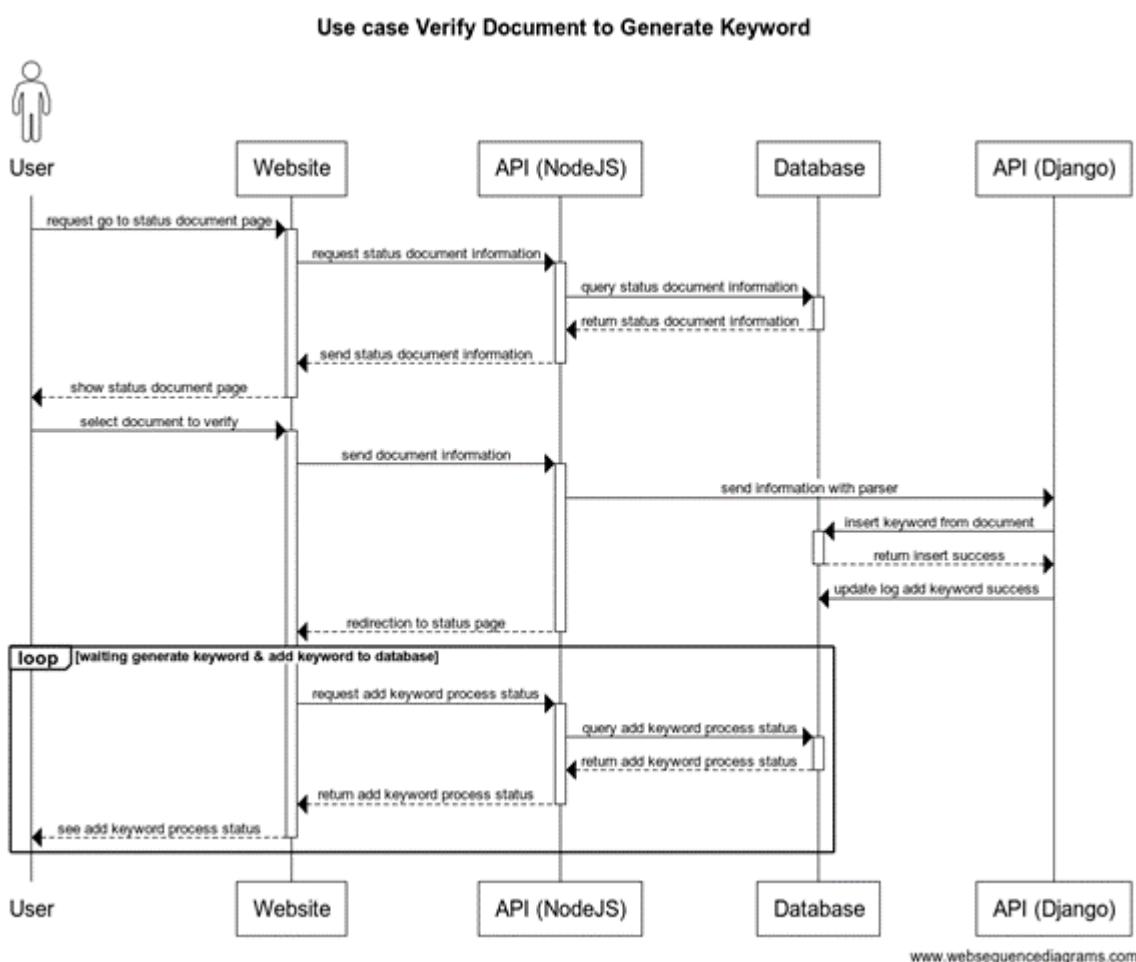
Scenario 3: ยืนยันสือว่าพร้อมสำหรับการถูกนำไปสร้างคีย์เวิร์ด

Goal: หนังสือถูกยืนยันพร้อมกับสร้างคีย์เวิร์ดเพื่อเพิ่มเข้าไปในระบบ

Precondition: ไปยังหน้าสถานะของหนังสือแล้วด้ไปยังปุ่มยืนยันหนังสือถูกต้อง

Main success scenario:

1. ผู้ใช้เข้าไปยังหน้าสถานะการเพิ่มหนังสือ
2. ระบบแสดงสถานะหนังสือว่าหนังสือไหนอยู่สถานะใดแล้วบ้าง
3. ผู้ใช้กดยืนยันว่าหนังสือถูกต้อง
4. ระบบย้ายไปหน้าสถานะหนังสืออีกครั้งเพื่อรอผลการทำงาน
5. ระบบแสดงการยืนยันหนังสือ และถูกเพื่อคีย์เวิร์ดเสร็จสนิท



รูปที่ 3.37 แสดง Scenario 3 ยืนยันสือว่าพร้อมสำหรับการถูกนำไปสร้างคีย์เวิร์ด

### 3.5.2.4 Use case Edit Document

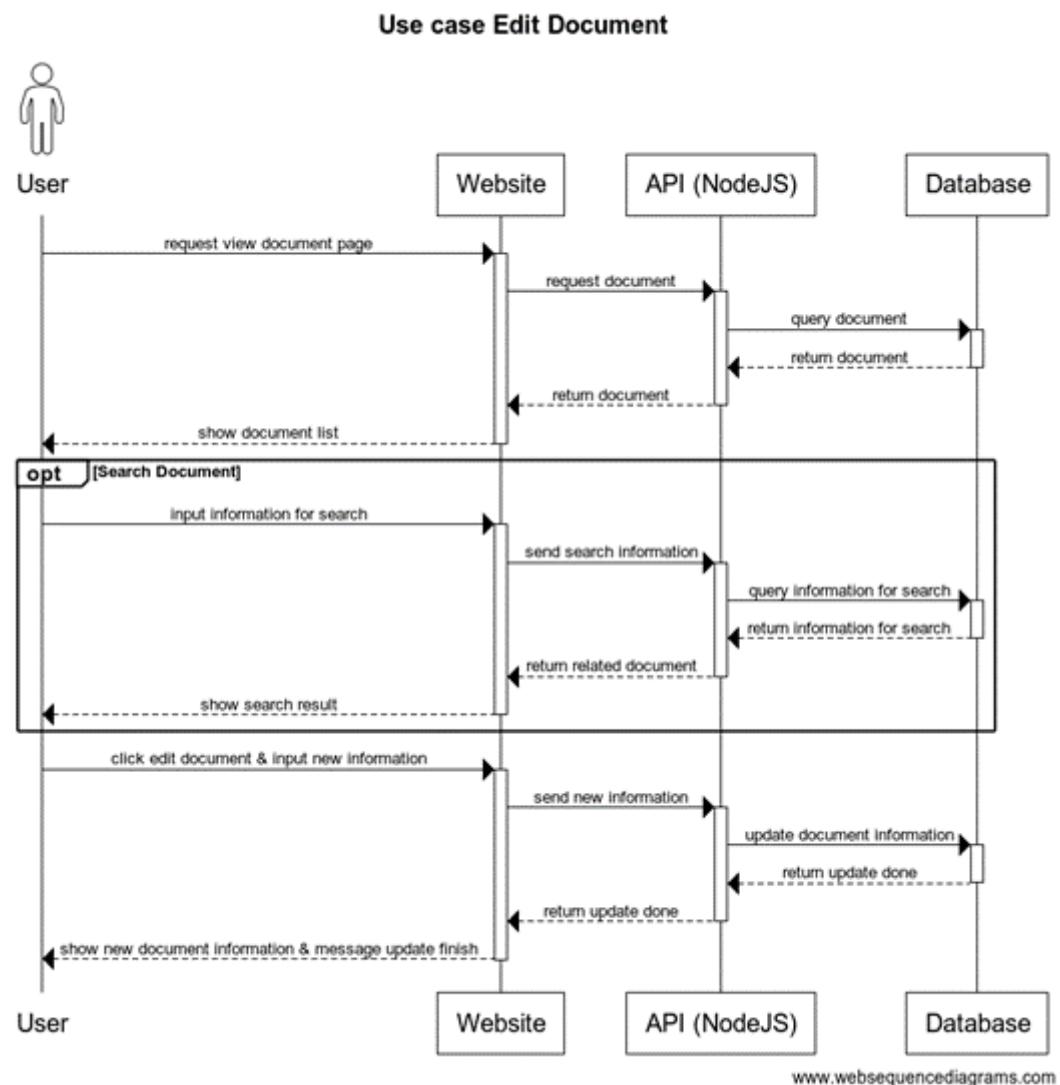
Scenario 4: การแก้ไขรายละเอียดของหนังสือ/หนังสือที่อยู่ภายในระบบ

Goal: รายละเอียดหนังสือถูกแก้ไขตามผู้ใช้งานต้องการ

Precondition: กดไปที่หัวข้อ MANAGE BOOK ใน Web Application

Main success scenario:

1. ผู้ใช้ค้นหาหนังสือที่ต้องการแก้ไขรายละเอียด
2. แสดงผลลัพธ์ในการค้นหาหนังสือ/หนังสือ
3. เลือกหนังสือ/หนังสือที่ต้องการแก้ไขรายละเอียด
4. แก้ไขรายละเอียดที่ต้องการ
5. กดบันทึกข้อมูลลงในระบบ



รูปที่ 3.38 แสดง Scenario 4 แก้ไขข้อมูลหนังสือ

### 3.5.2.5 Use case Delete Document

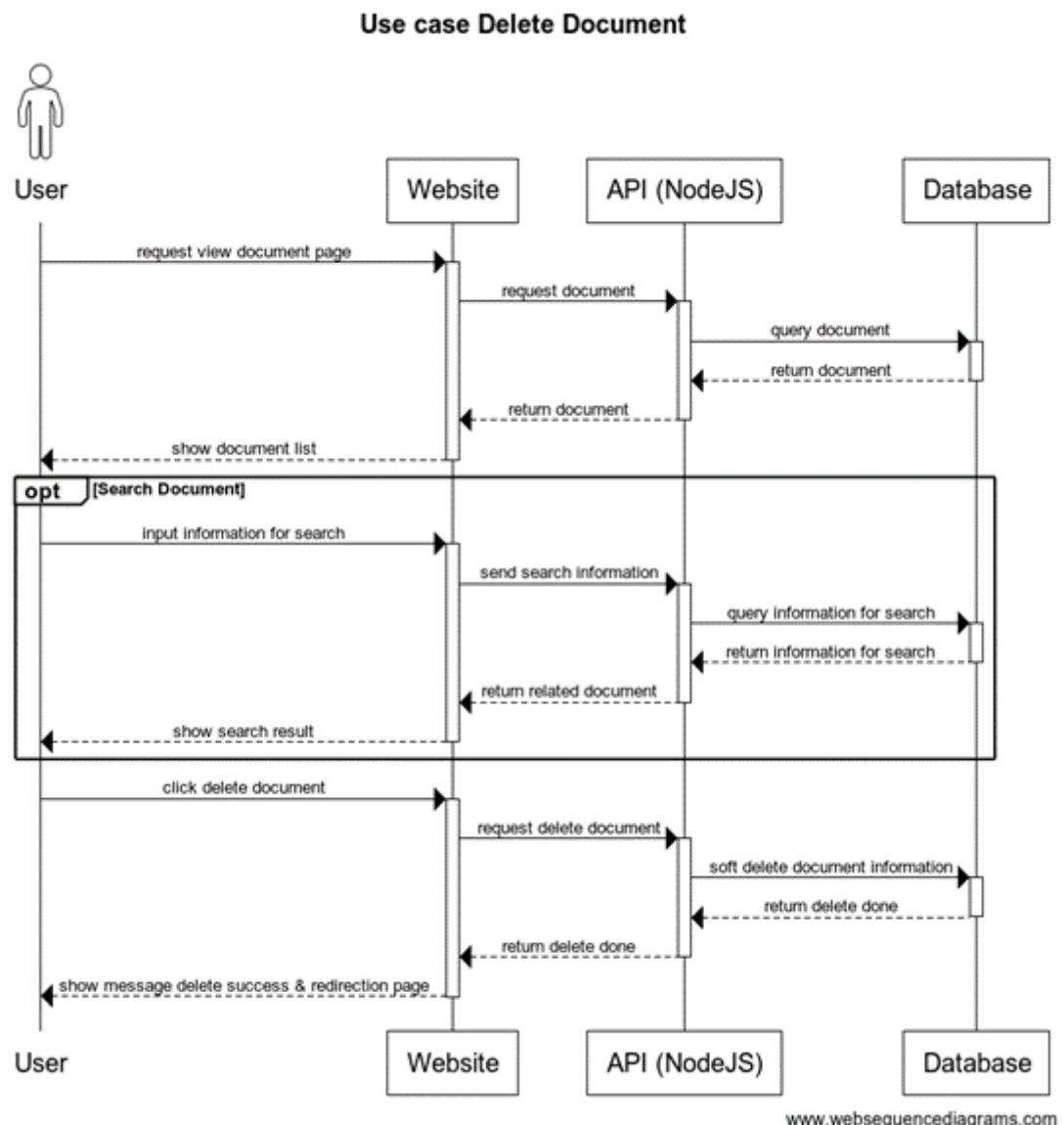
Scenario 5: ลบหนังสือ/หนังสือภายในระบบ

Goal: หนังสือ/หนังสือถูกนำออกจากระบบ

Precondition: กดเลือกหัวข้อ MANAGE BOOK ใน Web Application

Main success scenario:

1. ผู้ใช้ทำการค้นหาหนังสือหนังสือที่ต้องการจะลบออกจากระบบ
2. แสดงผลลัพธ์ในการค้นหาหนังสือ/หนังสือ
3. กดลบหนังสือ/หนังสือที่ต้องการ
4. กดยืนยันคำสั่งลบเพื่อบันทึกลงระบบ



รูปที่ 3.39 แสดง Scenario 5 ลบหนังสือ

### 3.5.2.6 Use case View Document & Search Document

Scenario 6: ดูข้อมูลหนังสือ และการค้นหาหนังสือ

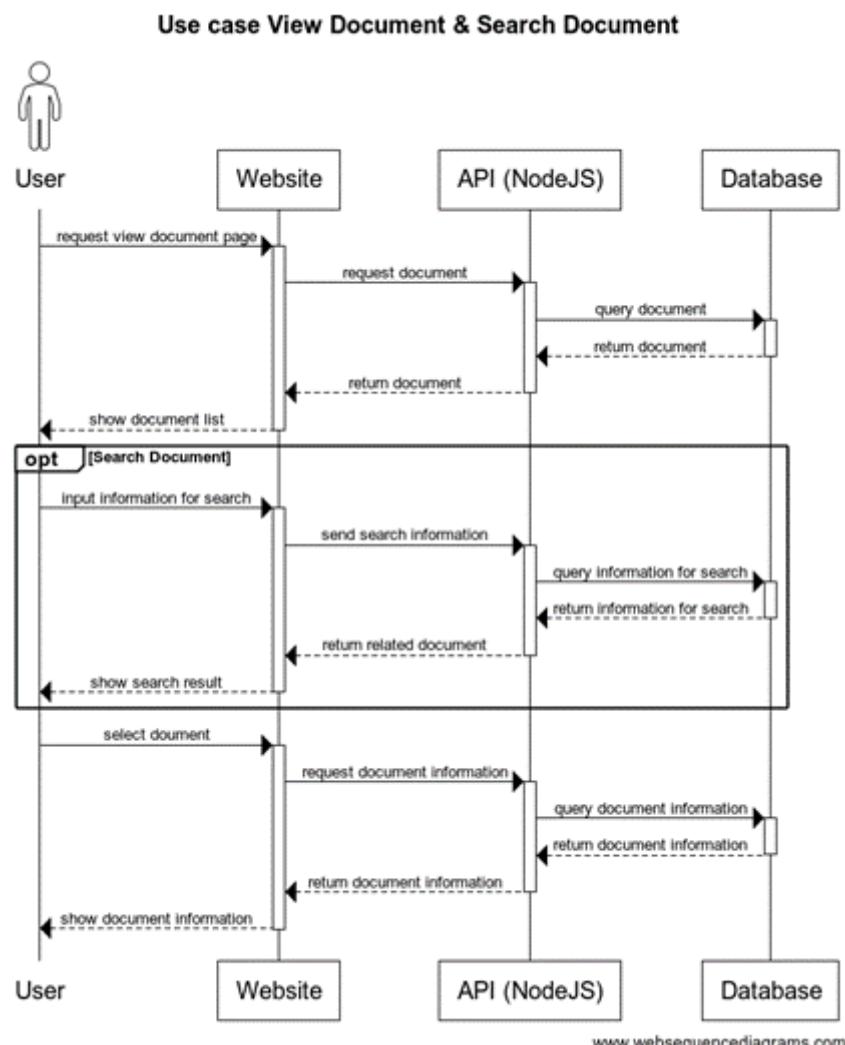
Goal: ผู้ใช้เจอหนังสือที่ต้องการ

Precondition: กดไปที่หัวข้อ SEARCH ใน Web Application

Main success scenario:

1. กรอกรายละเอียดข้อมูลที่ต้องการจะค้นหา
2. แสดงผลลัพธ์ในการค้นหา
3. ผู้ใช้เลือกหนังสือที่ต้องการที่จะดูข้อมูล

4. ระบบย้ายไปยังหน้าแสดงข้อมูลหนังสือที่ถูกเลือก



รูปที่ 3.40 แสดง Scenario 6 ดูข้อมูลหนังสือ และการค้นหาหนังสือ

### 3.5.2.7 Use case Login

Scenario 7: ระบบล็อกอิน

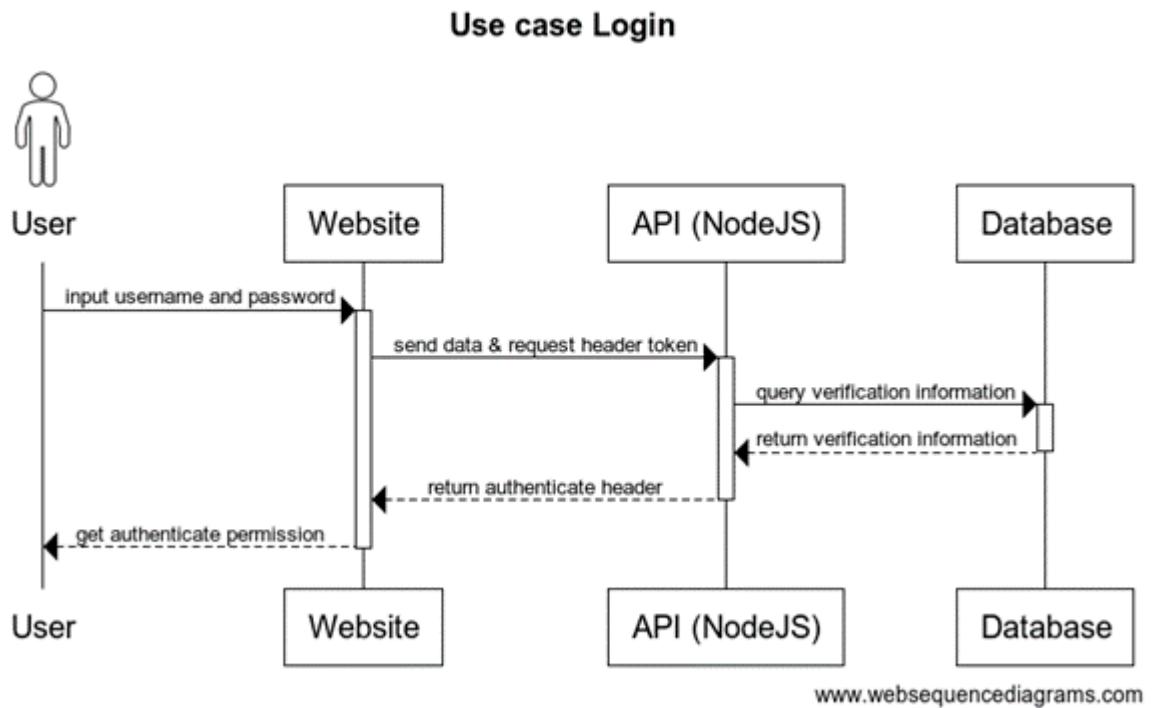
Goal: เพื่อเข้าสู่ระบบให้สามารถใช้ฟังก์ชันภายใน Web Application เพิ่มเติมได้

Precondition: กดหัวข้อ LOGIN ใน Web Application

Main success scenario:

1. ผู้ใช้กรอกชื่อผู้ใช้งานและรหัสผ่าน
2. กดเข้าสู่ระบบ
3. เข้าสู่ระบบสำเร็จ ส่งผู้ใช้กลับไปสู่ Homepage

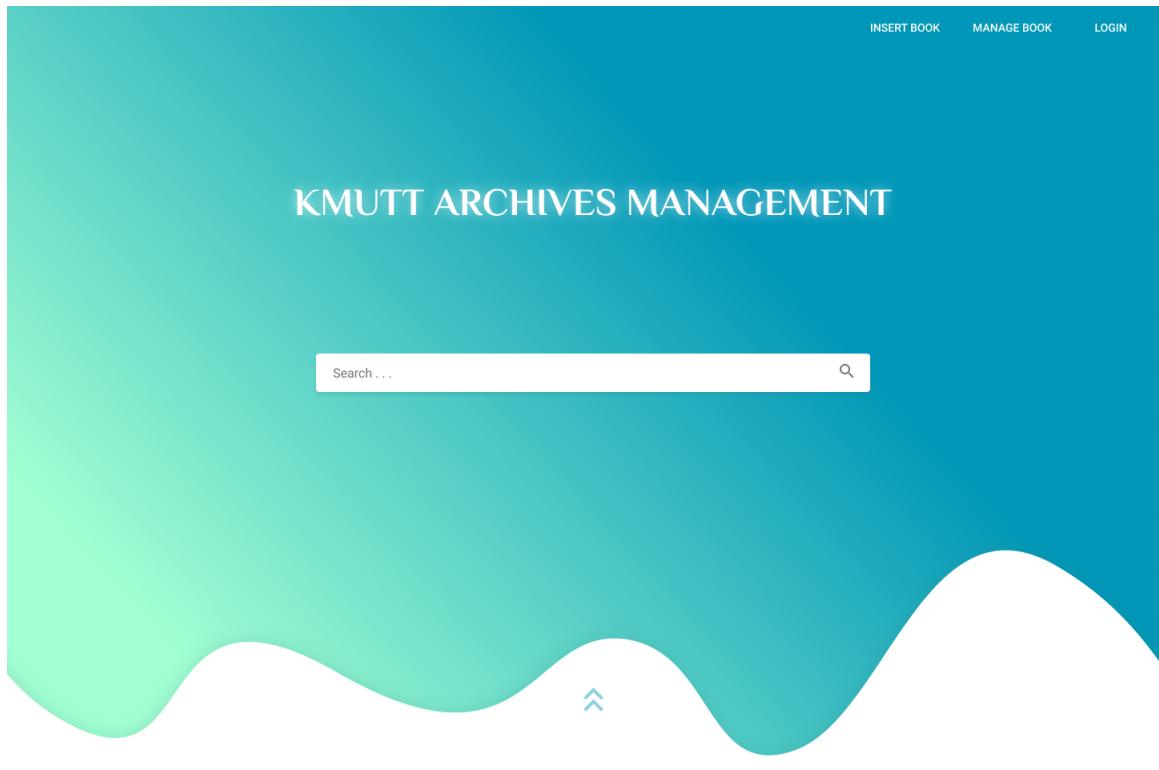
4. สามารถเข้าใช้งานฟังก์ชันของ Web Application ได้



รูปที่ 3.41 แสดง Scenario 7 ระบบล็อกอิน

### 3.6 GUI Design

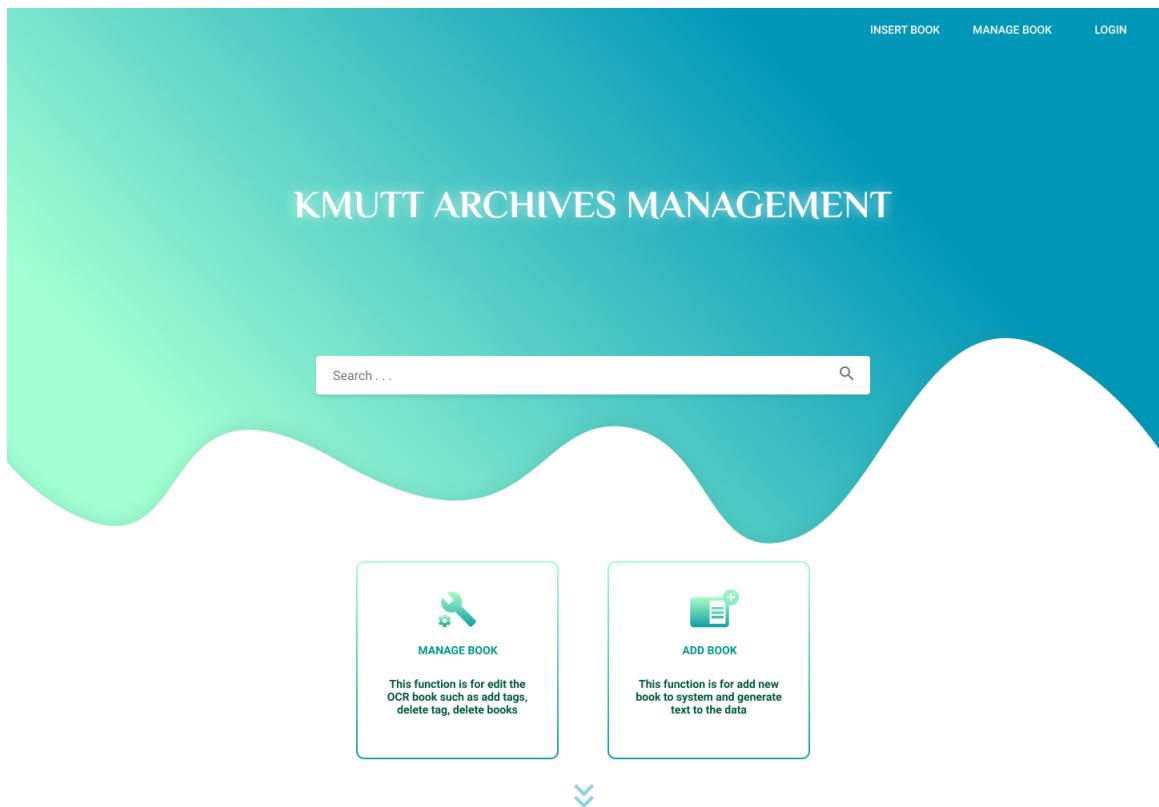
#### 3.6.1 Homepage



รูปที่ 3.42 ภาพแสดงหน้าหลักของเว็บไซต์

หน้าหลักของเว็บไซต์จะเป็นหน้าที่เน้นการค้นหาเป็นหลัก ที่ผู้ใช้สามารถเข้าถึงเมนูการเพิ่มหนังสือ การจัดการ และการเข้าสู่ระบบได้ที่  
แถบ Navigation ด้านบนของเว็บไซต์ดังรูปที่ 3.42

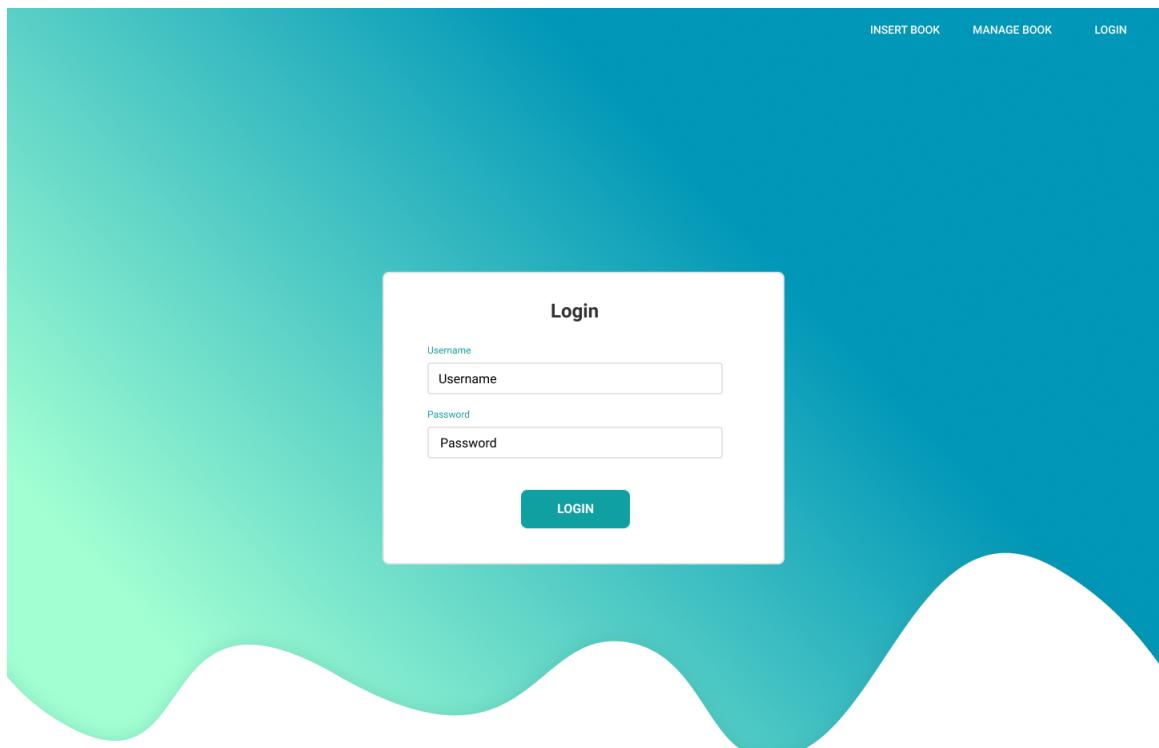
### 3.6.2 Homepage2



รูปที่ 3.43 ภาพแสดงหน้าหลักของเว็บไซต์หลังจากการกดเปิดเมนู

เมื่อกดปุ่มลูกศรที่ด้านล่างของรูป 3.42 จะมีเมนูเพิ่มเติมขึ้นมาคล้ายเป็นรูปที่ 3.43 ซึ่งจะแสดงรายละเอียดในแต่ละฟังก์ชันเพิ่มเติม

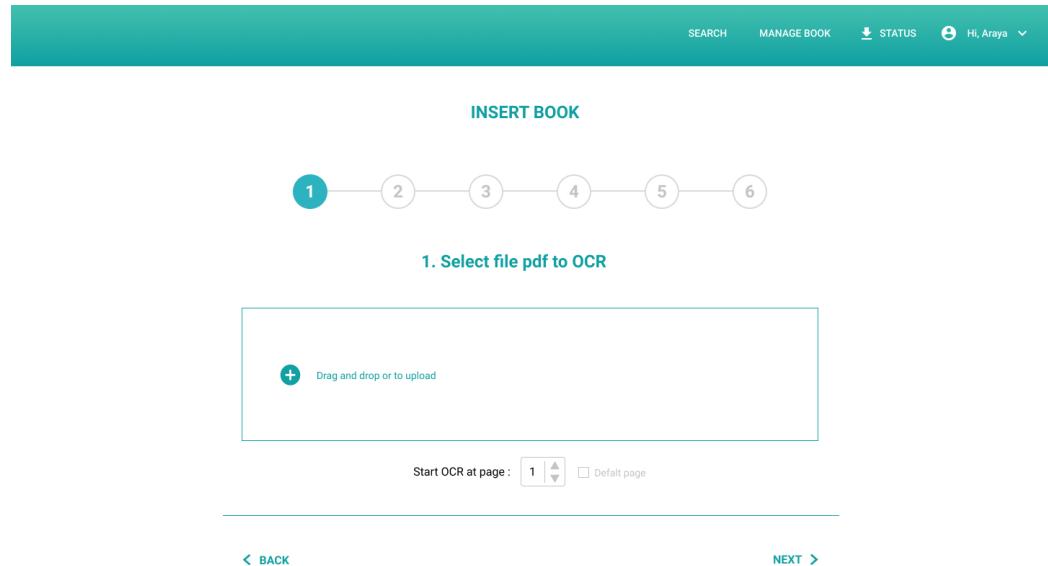
### 3.6.3 Login



รูปที่ 3.44 ภาพแสดงหน้าเข้าสู่ระบบ

ก่อนที่จะทำการเพิ่มหนังสือหรือจัดการกับหนังสือผู้ใช้นั้น จะต้องเข้าสู่ระบบก่อนเสมอ ถ้าเกิดกดเข้าฟังก์ชันการเพิ่มหนังสือหรือค้นหาโดยที่ยังไม่ได้เข้าสู่ระบบ ระบบจะบังคับให้ผู้ใช้เข้ามาในหน้าเข้าสู่ระบบดังรูป 3.25 เพื่อทำการเข้าสู่ระบบหรือจะเข้ามาโดยการกด log in ที่ปุ่มขวาบนได้

### 3.6.4 Insert Book(1)



รูปที่ 3.45 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นเลือกไฟล์

หน้าเพิ่มหนังสือขึ้นแรกจะเป็นการเลือกไฟล์หนังสือที่ต้องการโดยที่จะมีส่วนของการเพิ่มไฟล์ที่อยู่รูปของ pdf เพื่อทำ OCR จากนั้นจะสามารถเลือกได้ว่าจะทำการ OCR ตั้งแต่หน้าไหนดังรูปที่ 3.26

### 3.6.5 Insert Book (2)

**2. Fill the data**

**Title**

- Type \*
- Title Alternative
- Input

**Creator**

- Creator name
- Creator Organization name
- Input

**Description**

- Table of contents
- Summary
- Abstract
- Note

**Publisher**

- Publisher
- Publisher E-mail

**Contributor**

- Contributor
- Contributor Role

**Date**

- Date
- DD/MM/YY

**Coverage**

- Coverage Spatial
- Coverage Temporal
- YYYY

**Rights**

- Rights
- Rights Access

**BACK** **NEXT >**

รูปที่ 3.46 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 1

หน้าเพิ่มหนังสือขั้นตอนที่ 2 เป็นหน้าที่ต้องใส่ข้อมูลที่จำเป็นของหนังสือ โดยที่จำเป็นต้องใส่จะมีสัญลักษณ์กำกับไว้หรือกีดขวางหนังสือตั้ง รูป 3.27 โดยในหน้านี้จะมีกล่องใส่ข้อมูลที่ถูกกรอกบ่อย ๆ สำหรับผู้ใช้(เจ้าหน้าที่)

### 3.6.6 Insert Book (3)

**INSERT BOOK**

1 2 3 4 5 6

**3. Optional data**

**Identifier**

Identifier URL  
Input

Identifier ISBN  
Input

**Source**

Source  
Input

**Relation**

Relation + ADD  
Input

**Thesis**

Degree name  
Input

Degree level  
Input

Degree discipline  
Input

Degree grantor  
Input

**Type**

Type  
Text

**Language**

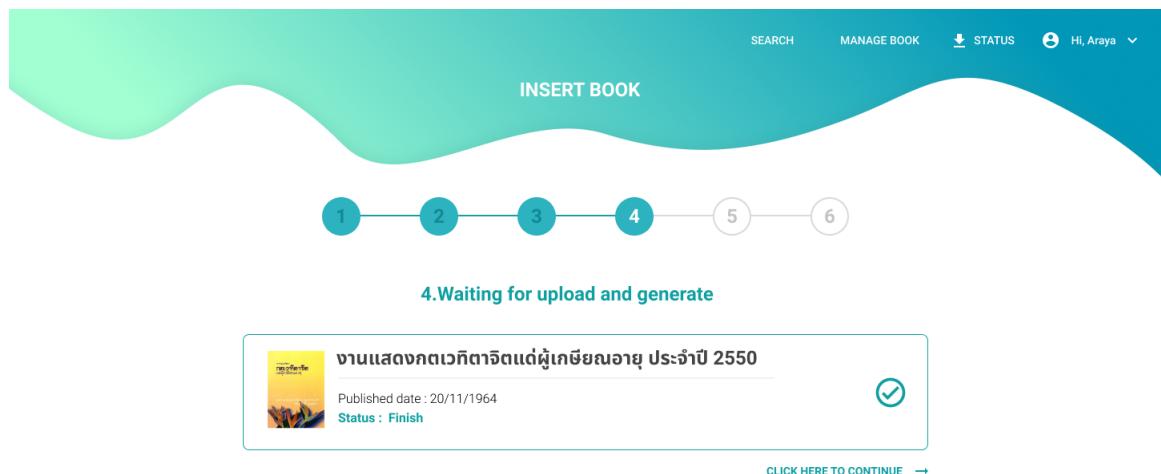
Language  
Thai

< BACK      NEXT >

รูปที่ 3.47 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 2

ในขั้นตอนที่ 3 จากรูปที่ 3.28 จะเป็นหน้าที่ใส่ข้อมูลที่ส่วนใหญ่จะไม่ต้องกรอกมากนัก ซึ่งไม่มีกล่องข้อมูลให้เห็นจำเป็นที่ต้องกรอกผู้ใช้สามารถข้ามไปขั้นตอนถัดไปได้เลย

### 3.6.7 Insert Book (4)



รูปที่ 3.48 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าขั้นโหลดข้อมูลเข้าสู่ระบบ

หลังจากที่ทำการใส่ข้อมูลอักษรทั้งหมดแล้วมาถึงหน้าที่เป็นหน้าโหลดข้อมูลตั้งรูป 3.29 ที่ระบบจะทำการ OCR และทำการเตรียมขุดข้อมูลที่ได้จากการ OCR โดยการนำคำมาตัดและเข็คคำผิด เมื่อโหลดข้อมูลเสร็จแล้วระบบจะทำการเปลี่ยนสถานะการโหลดและขึ้นล็อปเพื่อเข้าสู่ขั้นตอนถัดไปได้

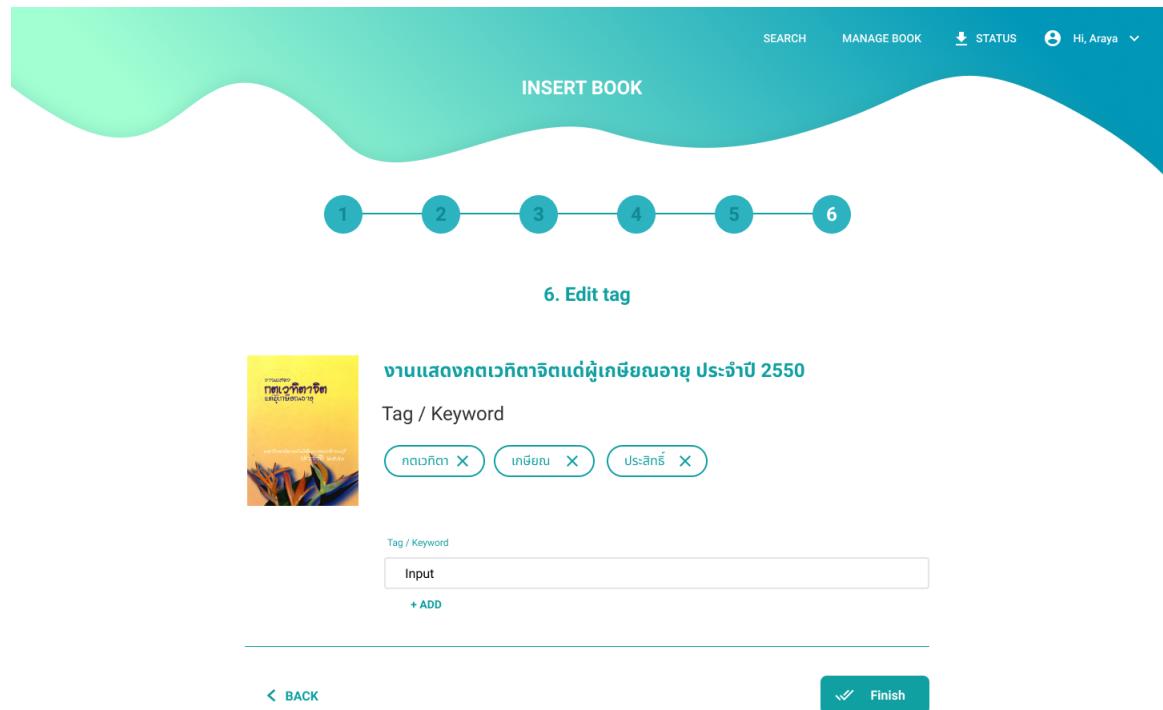
### 3.6.8 Insert Book (5)



รูปที่ 3.49 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นแก้ไขคำผิด

หลังจากโหลดและเตรียมข้อมูลเรียบร้อยแล้ว ระบบจะทำการแสดงข้อมูลที่ถูกแปลงมาโดยที่ผู้ใช้สามารถแก้ไขคำได้ดังรูป 3.30 หรือสามารถข้ามได้โดยใช้คีย์ไปที่กล่องข้อความจะขึ้นให้แก้แต่ละคำและเมื่อเปลี่ยนหน้าจะทำการเก็บข้อมูลที่เปลี่ยนไว้ และจะบันทึกการแก้ไขข้อมูลทั้งหมดที่แก้มีข้ามไปขั้นตอนตัดไป

### 3.6.9 Insert Book (6)



รูปที่ 3.50 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นแก้ไขและเพิ่มคำสำคัญ

หน้าสุดท้ายของการเพิ่มหนังสือจะเป็นหน้าที่ให้ผู้ใช้สามารถจัดการกับ Keyword ได้ดังรูปที่ 3.31 โดยเมื่อผู้ใช้ต้องการใส่คำสำคัญเพิ่มสามารถกด ADD เพื่อเพิ่มคำที่ต้องการใส่ได้ และสามารถลบเมื่อคลิกที่ปุ่มลบหากที่คำสำคัญที่ระบบทำการสร้างมาให้ เมื่อแก้ไขเสร็จแล้วสามารถกดปุ่ม Finish เพื่อทำการบันทึกข้อมูล

### 3.6.10 Search

The screenshot shows a library search interface with a teal header bar. The header includes 'EDIT BOOK', 'MANAGE BOOK', 'STATUS', and a user profile 'Hi, Araya'. A search bar at the top has the placeholder 'Search ...' and a magnifying glass icon. To the right of the search bar is a 'Filter' section with several checkboxes and an 'APPLY' button.

**Search results : KMUTT**

**งานแสดงกตเวกิตาจิตแด่ผู้เกียรติวุฒิ ประจำปี 2550**

Creator : Joe, Bryan  
Coverage temporal : 1998  
Tag : กิตติวุฒิ, เกียรติ, 1964

**งานแสดงกตเวกิตาจิตแด่ผู้เกียรติวุฒิ ประจำปี 2550**

Creator : Joe, Bryan  
Coverage temporal : 1998  
Tag : กิตติวุฒิ, เกียรติ, 1964

**งานแสดงกตเวกิตาจิตแด่ผู้เกียรติวุฒิ ประจำปี 2550**

Creator : Joe, Bryan  
Coverage temporal : 1998  
Tag : กิตติวุฒิ, เกียรติ, 1964

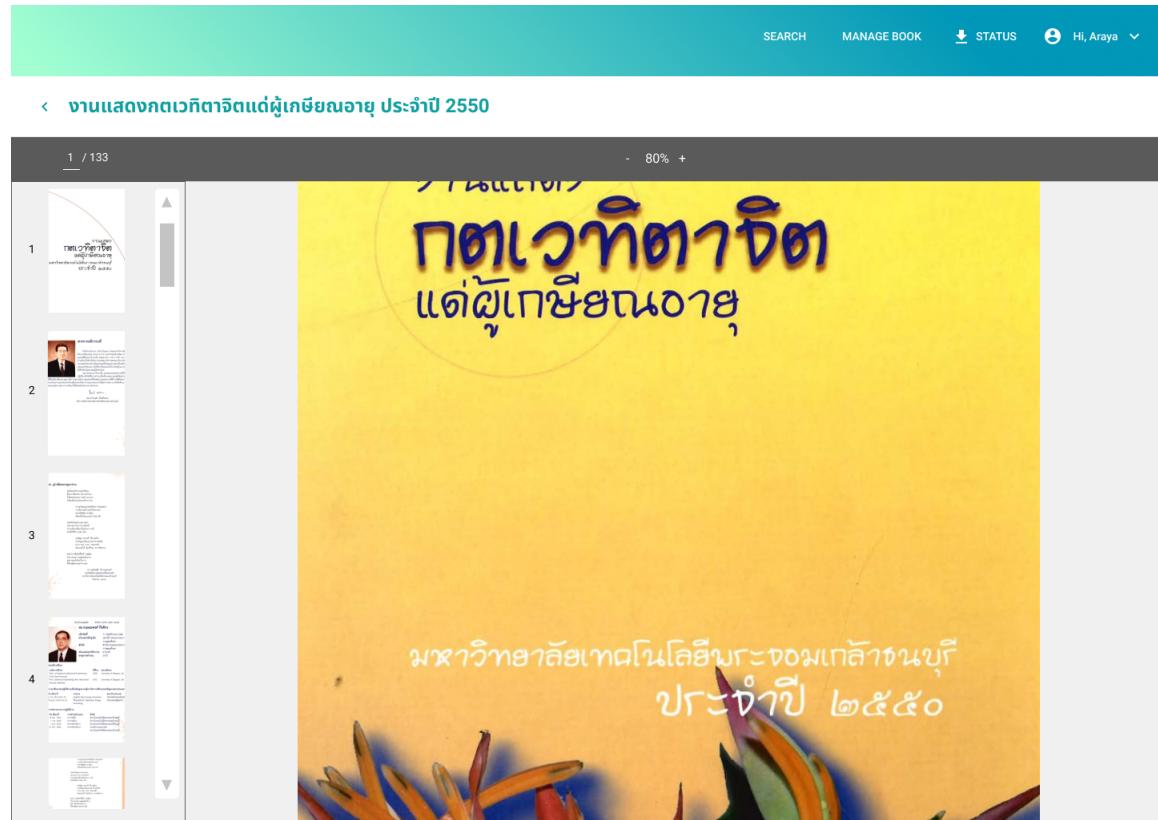
**งานแสดงกตเวกิตาจิตแด่ผู้เกียรติวุฒิ ประจำปี 2550**

Creator : Joe, Bryan  
Coverage temporal : 1998  
Tag : กิตติวุฒิ, เกียรติ, 1964

รูปที่ 3.51 ภาพแสดงหน้าค้นหาข้อมูล

หน้าแสดงข้อมูลการค้นหาเมื่อทำการค้นหาข้อมูลจากหน้าแรก (รูปที่ 3.23 หรือ 3.24) จะทำการแสดงข้อมูลหนังสือที่ตรงกับ keyword โดยเรียงคะแนนของหนังสือที่เกี่ยวข้องกับคำค้นมากที่สุดดังรูปที่ 3.32 เมื่อกดเข้าไปที่รายชื่อหนังสือจะทำการนำทางผู้ใช้ไปยังหน้าดูหนังสือดังรูปที่ 3.33

### 3.6.11 Document View



รูปที่ 3.52 ภาพแสดงหน้าดูหนังสือ

เมื่อเราค้นหาและเลือกหนังสือ ก็จะมีหน้าหนังสือ (รูปที่ 3.33) ขึ้นมาให้ดูเนื้อหาภายในโดยที่ผู้ใช้สามารถปรับขนาดภาพและสามารถเลือกหน้าที่ต้องการจะเปิดได้และสามารถย้อนหลับไปยังหน้าเดิมได้ที่ปุ่มลูกศรทางด้านซ้ายบน

### 3.6.12 Manage book

Search results : KMUTT

Actions	Title	Creator	Coverage temporal	Tags
<a href="#">DELETE</a>	งานแสดงกตเวกิตาอิตเด่อผู้เกจยนอยุ ประจำปี 2550	Creator : Joe, Bryan	Coverage temporal : 1998	Tag : กตเวกิตา, เกจยน, 1994
<a href="#">Edit</a>	งานแสดงกตเวกิตาอิตเด่อผู้เกจยนอยุ ประจำปี 2550	Creator : Joe, Bryan	Coverage temporal : 1998	Tag : กตเวกิตา, เกจยน, 1994
<a href="#">DELETE</a>	งานแสดงกตเวกิตาอิตเด่อผู้เกจยนอยุ ประจำปี 2550	Creator : Joe, Bryan	Coverage temporal : 1998	Tag : กตเวกิตา, เกจยน, 1994
<a href="#">Edit</a>	งานแสดงกตเวกิตาอิตเด่อผู้เกจยนอยุ ประจำปี 2550	Creator : Joe, Bryan	Coverage temporal : 1998	Tag : กตเวกิตา, เกจยน, 1994

รูปที่ 3.53 ภาพแสดงหน้าการจัดการหนังสือที่เพิ่มเข้าสู่ระบบ

ในหน้าของการจัดการหนังสือดังรูปที่ 3.34 จะมีลักษณะคล้ายกับหน้าการค้นหาเพียงแต่ว่าจะมีฟังก์ชันสำหรับการแก้ไขเนื้อหานะสือภายในที่ผู้ใช้เคยกรอกไว้ตอน OCR หนังสือมา เมื่อกดปุ่มลบจะมีหน้าต่างแจ้งเตือนเพื่อถามความแนใจในการลบหนังสือ หรือกดปุ่ม Edit เพื่อทำการเข้าสู่การแก้ไขข้อมูลของหนังสือนั้นๆดังรูปที่ 3.35 - 3.37

### 3.6.13 Edit Book

SEARCH MANAGE BOOK STATUS Hi, Page

**INSERT BOOK**

งานแสดงออกทางศิลปะสู่ภูมิประเทศ ประจวบ 2550

Creator: Joe Bryan  
Coverage Spatial: 1998  
Tag:

**Title**  
Title:   
Title Alternative:   
**Creator**  
Creator Name:   
Creator Organization Name:   
**Description**  
Title or Content:   
Summary:   
Abstract:   
Note:   
**Publisher**  
Publisher: มหาวิทยาลัยเทคโนโลยีราชมงคลเชียงใหม่  
Publisher Email:   
**Contributor**  
Contributor:   
Contributor Role:   
**Date**  
Issued Date:   
**Coverage**  
Coverage Spatial:   
Coverage Temporal:  YYYY  
**Rights**  
Rights:   
Rights Access: Available

< BACK **SAVE** NEXT >

รูปที่ 3.54 ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 1

**INSERT BOOK**

---



**งานแสดงผลเวกเตอร์ด้วยภาษา C++ ประจําปี 2550**

Creator : Joe, Bryan  
Coverage temporal : 1998  
Tag : ภาษาไทย หนังสือ 1964

---

**Identifier**

Identifier URL

Identifier ISBN

---

**Source**

Source

---

**Relation**

Relation + ADD

---

**Thesis**

Degree name

Degree level

Degree discipline

Degree grantor

---

**Type**

Type Text

---

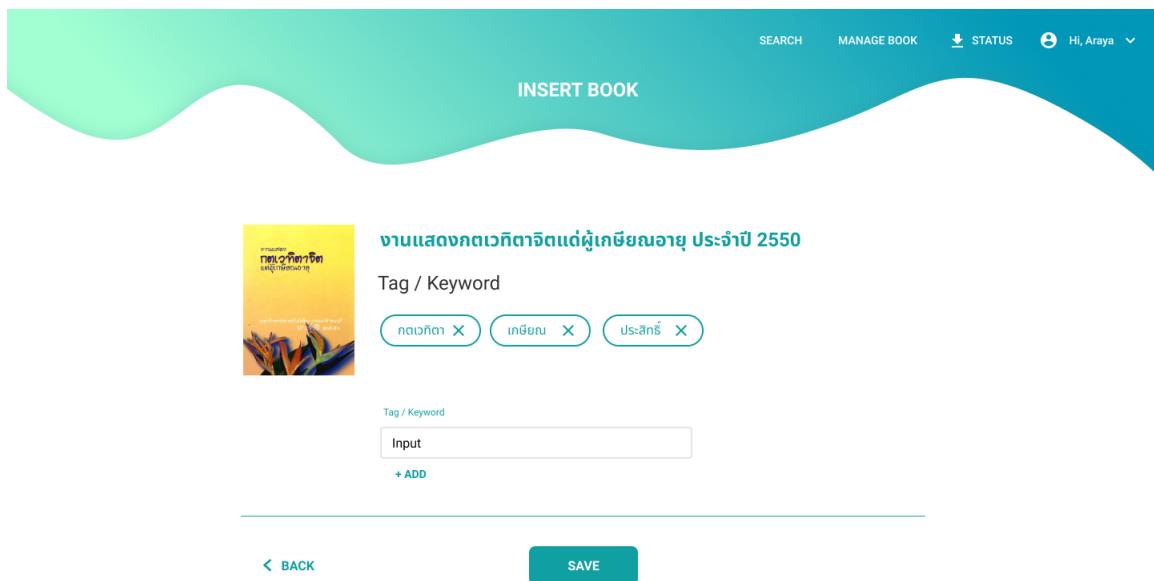
**Language**

Language Thai

---

< BACK SAVE NEXT >

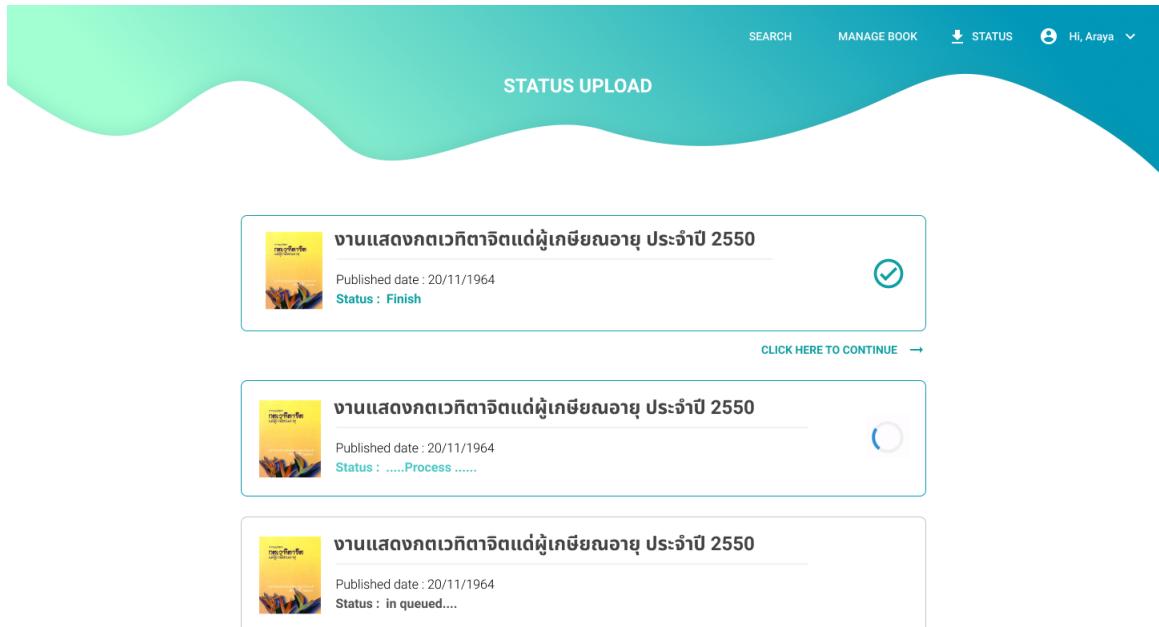
รูปที่ 3.55 ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 2



รูปที่ 3.56 ภาพแสดงขั้นตอนการแก้ไขหนังสือขั้นที่ 3

หน้าแก้ไขหนังสือแบ่งออกเป็น 3 ขั้นตอนดังรูป 3.35 - 3.37 ซึ่งจะมีให้แก้ไข ข้อมูลที่เคยกรอกไว้ตอนเพิ่มนั้นงสือเข้ามา โดยจะมีรูปปักหนังสือและชื่อหนังสืออยู่บอกว่ากำลังแก้ไขหนังสือเล่มใหม่อยู่ และในทุกหน้าจะมีปุ่มสำหรับบันทึกในทุกหน้าเพื่อที่จะสามารถบันทึกโดยที่ไม่ต้องรอไปหน้าสุดท้ายเพื่อบันทึกข้อมูล

### 3.6.14 Upload Status Page



รูปที่ 3.57 ภาพแสดงหน้าการโหลดข้อมูล

จากรูป 3.38 สำหรับผู้ใช้ที่ทำการเพิ่มหนังสือเข้าสู่ระบบจะมีหน้าสำหรับโหลดกรณีที่กดออกมากลับจากผ่านขั้นตอนการเพิ่มหนังสือขั้นตอนที่ 4 จะสามารถเข้ามาดูสถานะและทำการดำเนินการต่อได้โดยไม่ต้องผ่านการเพิ่มหนังสือเข้าสู่ระบบใหม่

### 3.6.15 Evaluate Process Design

ในส่วนของการประเมินผลการทำงานนี้จะแบ่งออกเป็น 3 ส่วนคือการออกแบบ User Interface ส่วนของการเตรียมข้อมูลรูปภาพ จะช่วยให้การทำ OCR มีประสิทธิภาพมากเท่าไร และส่วนของระบบการค้นหา โดยในส่วนของ OCR จะทำการประเมินจากการเลือกเช็คคำจาก 2 หน้าของแต่ละหนังสือมาเช็คว่าแต่ละหน้ามีคำผิดเท่าไร โดยจะเลือกวัดหนังสือทั้งหมด 5 เล่มแบบสุ่มและเทียบการเตรียมข้อมูลรูปภาพ ว่าทำแบบไหนได้ผลลัพธ์แบบไหนอ่อนกว่า

ตารางที่ 3.21 ตารางประเมินการทำ OCR

ตารางประเมินการทำ OCR				
หนังสือ	หน้า	จำนวนคำทั้งหมด	คำที่ผิด(%)	คำเกิน(คำ)

ระบบการค้นหา จะเช็คโดยให้ผู้ใช้เป็นผู้ประเมินว่าได้รับหนังสือตรงตามที่ต้องการหรือไม่โดยจะให้เจ้าหน้าที่บรรณาธิการคัดเลือกหนังสือจำนวน 3 เล่มที่คาดหวังว่าจะขึ้นมาเมื่อค้นหาทั้งหมด 10 ครั้ง

ตารางที่ 3.22 ตารางประเมินระบบการค้นหา

ตารางประเมินระบบการค้นหา		
คำค้นหา	หนังสือที่คาดหวัง	การค้นหา
		<p>คะแนน 5 ระดับ</p> <p>5 = ค้นหาหนังสือได้ตรงตามที่ต้องการ และมีหนังสือที่เกี่ยวข้องกับคำค้นหาขึ้นมาอย่างถูกต้อง</p> <p>4 = ค้นหาหนังสือได้ถูกต้องตามที่ต้องการ บางเล่มและมีหนังสือที่เกี่ยวข้องกับคำค้นหาขึ้นมา</p> <p>3 = ไม่สามารถค้นหาหนังสือที่ต้องการแต่ มีหนังสือที่เกี่ยวข้องกับคำค้นหาขึ้นมา</p> <p>2 = สามารถค้นหาหนังสือที่มีความเกี่ยวข้องกับคำค้นหา และมีหนังสือที่ไม่เกี่ยวข้องกับการค้นหาแสดงในผลลัพธ์</p> <p>1 = ไม่มีหนังสือที่เกี่ยวข้องขึ้นมาในผลลัพธ์</p>

ตารางที่ 3.23 ตารางประเมินความพึงพอใจการออกแบบ UX/UI

ตารางประเมินการออกแบบ UX/UI					
	4	3	2	1	คะแนนที่ได้
ความ สมบูรณ์ ของ ข้อมูล	ข้อมูล มี ความ สมบูรณ์ ชัดเจน ทำให้ เข้าใจ ความ หมาย ที่ ต้องการจะสื่อได้เป็น อย่างดี	มี ข้อมูล ที่ ชัดเจน และ แม่นยำ ใน บาง ครั้ง และ สามารถ แสดง ความ หมาย ที่ ต้องการจะสื่อได้บ้าง	ข้อมูลความแม่นยำ และชัดเจนบ้าง	มี ข้อมูล ที่ ไม่ ชัด- เจน ไม่ครบ สื่อความ หมายได้ไม่ดี	คะแนนที่ได้
การออกแบบ	มี การ ออกแบบ ที่ เน้น ความ สำคัญ และ จัด วาง องค์ ประกอบ สี เสียง และ การ เคลื่อน- ไหว(animation) ได้ อย่างเหมาะสม	มี การ จัด หน้า และ องค์ ประกอบ ทำให้ เห็นใจ ความ สำคัญ ของ เนื้อหา มี การ ใช้ การ เคลื่อน- ไหว(animation) บ้าง	การวางหน้าและการ จัด องค์ ประกอบ มี ความไม่เหมาะสม มี การ ใช้ การ เคลื่อน- ไหว(animation) เข้ามาช่วยบ้าง	การ วาง หน้า และ การจัดองค์ประกอบ มี ความ ไม่ เหมาะ สม และ ไม่มี การ ใช้ การ เคลื่อน- ไหว(animation) เข้ามาช่วยในการใช้ งาน	
การใช้งาน	ผู้ใช้ สามารถ ใช้งาน ปุ่ม หรือ ย้าย ไป ยัง หน้า ต่างๆ ได้ อย่าง ง่ายดาย แต่มีลิ้งค์ที่ พา ไป ผิด หน้า อย่าง มาก หนึ่ง ลิ้งค์ หรือ ไม่มีเลย	ผู้ใช้ สามารถ ใช้งาน ปุ่ม หรือ ย้าย ไป ยัง หน้า ต่างๆ ได้ อย่าง ง่ายดาย แต่มีลิ้งค์ที่ พา ไป ผิด หน้า อย่าง มากสองลิ้งค์	ผู้ใช้มีความสับสนใน การ ใช้ ปุ่ม หรือ การ ย้าย ไป ยัง หน้า ต่างๆ บาง ครั้ง และ มีลิ้งค์ ที่ พา ไป ผิด หน้า อย่าง มากสามลิ้งค์	ผู้ใช้เกิดความสับสน ใน ปุ่ม หรือ ลิ้งค์ ที่ ย้าย ไป หน้า ต่างๆ	
การใช้ภาษา	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ ไม่ เหมาะ สม อาย่างมาก 1 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ ไม่ เหมาะ สม อาย่างมาก 2 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ ไม่ เหมาะ สม อาย่างมาก 3 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ ไม่ เหมาะ สม มา กกว่า 4 จุด	

ตารางที่ 3.24 ตารางประเมินการทำสอบเว็บไซต์

เกณฑ์การประเมิน	ผลลัพธ์		หมายเหตุ
	ผ่าน	ไม่ผ่าน	
1. สามารถเข้าสู่ระบบและออกจาก ระบบได้			
2. สามารถเพิ่มหนังสือเข้าสู่ระบบได้			
3. สามารถแก้ไขรายละเอียดหนังสือที่อยู่ในระบบได้			
4. สามารถตรวจสอบและแก้ไขคำที่เพิ่มเข้ามาในระบบในขั้นตอนเพิ่มหนังสือได้			
5. สามารถลบหนังสือที่อยู่ในระบบได้			
6. สามารถค้นหาข้อมูลหนังสือภายในระบบได้			
7. สามารถเรียกดูหนังสือที่ต้องการได้			

## บทที่ 4 ผลการดำเนินงาน

ดำเนินงานของโครงการนี้จะแบ่งออกมาเป็นทั้งหมด 3 ส่วน โดยส่วนแรกคือส่วนของการจัดเก็บข้อมูลเข้าสู่ระบบโดยนำรูปภาพได้ที่ได้รับมาผ่านกระบวนการการเตรียมข้อมูลรูปภาพ ก่อนจะนำไปผ่านกระบวนการ OCR และการเตรียมข้อมูลตัวหนังสือ ก่อนจะถูกเก็บข้อมูลในระบบ ส่วนที่สองการค้นหาข้อมูล เป็นการค้นหาแบบ IR (Information retrieval) โดยนำคำแนะนำ TF-IDF มาใช้เป็นคะแนนในการค้นหา และส่วนสุดท้ายคือส่วนของการทำแพลตฟอร์มเว็บไซต์

### 4.1 ผลลัพธ์ที่ได้จากการทำการเตรียมข้อมูลรูปภาพ

#### 4.1.1 ผลลัพธ์ที่ได้จากการประเมินภาพของหุ่น

ผลลัพธ์จากการหุ่นภาพตัวหนังสือทั้ง 978 ภาพ มีความคลาดเคลื่อนทั้งหมด 7.98% ที่ยังไม่สามารถหุ่นภาพให้ตรง และทำให้บางภาพแยก不清 เนื่องจากว่าบรรทัดตัวอักษรอาจมีเส้นที่ไม่สามารถทำกรวยกันให้ถูกต้องเป็นเส้นบรรทัดได้

#### 4.1.2 เปรียบเทียบประสิทธิภาพในการทำ OCR ของ การทำการเตรียมข้อมูลรูปภาพ แต่ละแบบ

จากการทดสอบประสิทธิภาพของการทำการเตรียมข้อมูลรูปภาพทั้งสองแบบพบว่า การทำการเตรียมข้อมูลรูปภาพ แบบแรกนั้นมีค่าผิดน้อยกว่า แต่มีค่าที่ไม่ถูกอ่านมากถึง 32.71% ดังตารางที่ [4.1](#) ซึ่งต่างจากการทำการเตรียมข้อมูลรูปภาพ แบบที่ 2 ที่ค่าความผิดต้องของคามี 74.74 % ดังตาราง [4.2](#)

##### 4.1.2.1 แบบที่ 1 การใช้การข้ามหน้า, การหุ่น, การลบรูปภาพ, การลบเส้น และการจัดกลุ่ม

ตารางที่ [4.1](#) ตารางประมวลผลการทำงานเตรียมข้อมูลรูปภาพแบบที่ 1

หนังสือ	หน้า	คำทั้งหมด	คำผิดที่เจอ	%	คำเกิน	คำที่ไม่โดนอ่าน	%
กตเวทิตาปี 2542	15	4	4	100 %	0	0	0 %
	29	252	14	5.56 %	46	2	0.79 %
กตเวทิตาปี 2556	15	242	33	13.64 %	2	1	0.41 %
	29	257	20	7.78 %	3	10	3.89 %
รายงานประจำปี 2544	15	47	3	6.38 %	2	34	72.34 %
	29	585	39	6.67 %	3	308	52.65 %
รายงานประจำปี 2553	15	68	0	0 %	0	68	100 %
	29	596	17	2.85 %	8	340	57.05 %
รายงานประจำปี 2549	15	155	53	34.19 %	42	45	29.03 %
	29	304	22	7.24 %	20	13	4.28 %
	total	2510	205	8.17 %	126	821	32.71 %

#### 4.1.2.2 แบบที่ 2 ใช้การลบพื้นหลัง

ตารางที่ 4.2 ตารางประเมินการทำการเตรียมข้อมูลรูปภาพแบบที่ 2

หนังสือ	หน้า	คำทั้งหมด	คำผิดที่เจอ	%	คำเกิน	คำที่ไม่โดนอ่าน	%
กตเวทิตาปี 2542	15	4	4	100%	0	0	0%
	29	252	30	11.9%	6	9	3.57%
กตเวทิตาปี 2556	15	242	42	17.36%	2	48	19.83%
	29	257	54	21.01%	2	62	24.12%
รายงานประจำปี 2544	15	47	27	57.45%	5	5	10.64%
	29	585	101	17.26%	23	0	0%
รายงานประจำปี 2553	15	68	30	44.12%	7	0	0%
	29	596	85	14.26%	30	0	0%
รายงานประจำปี 2549	15	155	57	36.77%	14	4	2.58%
	29	304	76	25%	7	0	0%
	total	2510	506	20.16%	96	128	5.1%

## 4.2 ผลการเปรียบเทียบข้อมูล 2 ชุด

จากการเปรียบข้อมูลทั้งสองชุด(ชุดที่ดีที่สุดชุดที่ 2)ดังตารางที่ 4.4 กับชุดข้อมูลชุดใหม่ล่าสุด(ชุดที่ 1)ดังตารางที่ 4.3 พบร่วมประสิทธิภาพของข้อมูลชุดที่ 1 นั้นมีคำผิดสูงกว่าประมาณ 2% เมื่อเทียบกับข้อมูลชุดที่ 2 แต่ว่ามีคำเกินที่ต่างกันเป็นเท่าตัว และมีคำที่ไม่โดนอ่านมากกว่า 28 คำ

ตารางที่ 4.3 ตารางประเมินข้อมูลชุดที่ 1

หนังสือ	หน้า	คำทั้งหมด	คำผิดที่เจอ	%	คำเกิน	คำที่ไม่โดนอ่าน	%
กตเวทิตาปี 2542	15	4	2	50%	0	2	50%
	29	252	34	13.49%	12	4	1.59%
กตเวทิตาปี 2556	15	242	37	15.29%	0	49	20.25%
	29	257	47	18.29%	2	45	17.51%
รายงานประจำปี 2544	15	47	40	85.11%	0	4	8.51%
	29	585	78	13.33%	11	15	2.56%
รายงานประจำปี 2553	15	68	44	64.71%	0	0	0%
	29	596	76	12.75%	9	12	2.01%
รายงานประจำปี 2549	15	155	44	28.39%	15	1	0.65%
	29	304	53	17.43%	34	0	0%
	total	2510	455	18.13%	83	132	5.26%

ตารางที่ 4.4 ตารางประเมินข้อมูลชุดที่ 2

หนังสือ	หน้า	คำทั้งหมด	คำพิเศษ	%	คำเกิน	คำที่ไม่ดูน่าอ่าน	%
กตเวทิตาปี 2542	15	4	4	100%	0	0	0%
	29	252	40	15.87%	20	10	3.97%
กตเวทิตาปี 2556	15	242	46	19.01%	11	44	18.18%
	29	257	32	12.45%	2	62	24.12%
รายงานประจำปี 2544	15	47	26	55.32%	0	4	8.51%
	29	585	63	10.77%	7	28	4.79%
รายงานประจำปี 2553	15	68	36	52.94%	9	2	2.94%
	29	596	65	10.91%	60	2	0.34%
รายงานประจำปี 2549	15	155	43	27.74%	30	8	5.16%
	29	304	52	17.11%	34	0	0%
	total	2510	407	16.22%	173	160	6.37%

### 4.3 ประสิทธิภาพการแก้ไขคำพิเศษ

จากการเปรียบข้อมูลที่ไม่ถูกแก้ไขคำพิเศษ [4.5](#) กับข้อมูลที่ผ่านระบบการแก้ไขคำพิเศษในตารางที่ [4.3](#) พบร่วมกันว่าการใช้ระบบการแก้คำพิเศษทำให้คำพิเศษที่เกิดขึ้นนั้นน้อยประมาณ 2%

ตารางที่ 4.5 ตารางประเมินข้อมูลชุดที่ 1 ที่ไม่ผ่านการแก้คำพิเศษ

หนังสือ	หน้า	คำทั้งหมด	คำพิเศษ	%	คำเกิน	คำที่ไม่ดูน่าอ่าน	%
กตเวทิตาปี 2542	15	4	4	100%	0	0	0%
	29	252	30	11.9%	6	9	3.57%
กตเวทิตาปี 2556	15	242	42	17.36%	2	48	19.83%
	29	257	54	21.01%	2	62	24.12%
รายงานประจำปี 2544	15	47	27	57.45%	5	5	10.64%
	29	585	101	17.26%	23	0	0%
รายงานประจำปี 2553	15	68	30	44.12%	7	0	0%
	29	596	85	14.26%	30	0	0%
รายงานประจำปี 2549	15	155	57	36.77%	14	4	2.58%
	29	304	76	25%	7	0	0%
	total	2510	506	20.16%	96	128	5.1%

#### 4.4 การประเมินความพึงพอใจของบรรณาธิการต่อการออกแบบ UX/UI

ตารางที่ 4.6 ตารางประเมินความพึงพอใจการออกแบบ UX/UI

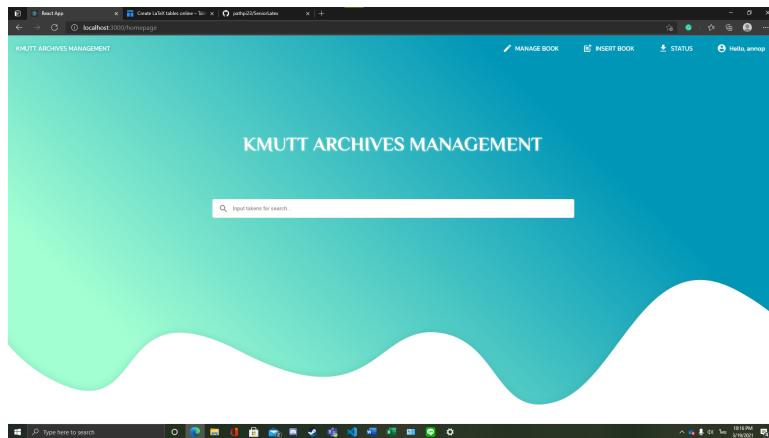
	4	3	2	1	คะแนนที่ได้
ความ สมบูรณ์ ของ ข้อมูล	ข้อมูล มี ความ สม- บูรณ์ ชัดเจน ทำให้ เข้าใจ ความ หมาย ที่ ต้องการจะสื่อได้เป็น อย่างดี	มี ข้อมูล ที่ ชัดเจน และ แม่นยำ ใน บาง ครั้ง และ สามารถ แสดง ความ หมาย ที่ ต้องการจะสื่อได้บ้าง	ข้อมูลมีความแม่นยำ และชัดเจนบ้าง	มี ข้อมูล ที่ 'ไม่' ชัด- เจน 'ไม่ครบ สื่อความ หมายได้ไม่ดี'	3
การออกแบบ	มี การ ออกแบบ ที่ เน้น ความ สำคัญ และ จัด วาง องค์ ประกอบ ศี ลี ยง และ การ เคลื่อน- ไหว(animation) ได้ อย่าง เหมาะสม	มี การ จัด หน้า และ องค์ ประกอบ ทำให้ เห็นใจ ความ สำคัญ ของ เนื้อหา มี การ ใช้ การ เคลื่อน- ไหว(animation) บ้าง	การ วาง หน้า และ การ จัด องค์ ประกอบ มี ความ 'ไม่' เหมาะ สม และ 'ไม่มี' การ ใช้ การ เคลื่อน- ไหว(animation) เข้ามาช่วยบ้าง	การ วาง หน้า และ การ จัด องค์ ประกอบ มี ความ 'ไม่' เหมาะ สม และ 'ไม่มี' การ ใช้ การ เคลื่อน- ไหว(animation) เข้ามาช่วยในการใช้ งาน	4
การใช้งาน	ผู้ใช้ สามารถ ใช้งาน ปุ่ม หรือ ย้าย ไป ยัง หน้า ต่างๆ ได้ อย่าง ง่ายดาย แต่มีลิ้งค์ที่ พาไป ผิด หน้า อย่าง มาก หนึ่ง ลิ้งค์ หรือ 'ไม่มีเลย'	ผู้ใช้ สามารถ ใช้งาน ปุ่ม หรือ ย้าย ไป ยัง หน้า ต่างๆ ได้ อย่าง ง่ายดาย แต่มีลิ้งค์ที่ พาไป ผิด หน้า อย่าง มากสองลิ้งค์	ผู้ใช้มีความสับสนใน การ ใช้ปุ่ม หรือ การ ย้าย ไป ยัง หน้า ต่างๆ บาง ครั้ง และ มีลิ้งค์ ที่ พาไป ผิด หน้า อย่าง มากสามลิ้งค์	ผู้ใช้ เกิด ความ สับสน ใน ปุ่ม หรือ ลิ้งค์ ที่ ย้าย ไป หน้า ต่างๆ	4
การใช้ภาษา	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ 'ไม่' เหมาะ สมอย่างมาก 1 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ 'ไม่' เหมาะ สมอย่างมาก 2 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ 'ไม่' เหมาะ สมอย่างมาก 3 จุด	มี การ ใช้ คำ ผิด หรือ ภาษา ที่ 'ไม่' เหมาะ สม มากกว่า 4 จุด	4

#### 4.5 ผลลัพธ์จากการค้นหา

ได้มีการออกแบบตารางสำหรับการวัดผลลัพธ์สำหรับการค้นหาทั้งรูปแบบธรรมดากลยุทธ์และการใช้โมเดล word2vec เข้ามาช่วยแต่เนื่องจาก การล่าช้าจึงทำให้ยังไม่มีการทดสอบผลลัพธ์จากผู้ใช้งานแต่มีการวางแผนว่าเมื่อการค้นหาเสร็จสิ้นจะดำเนินการวัดผลลัพธ์ที่เกิดขึ้นสำหรับ การค้นหาว่าตรงตามที่ผู้ใช้ต้องการหรือไม่

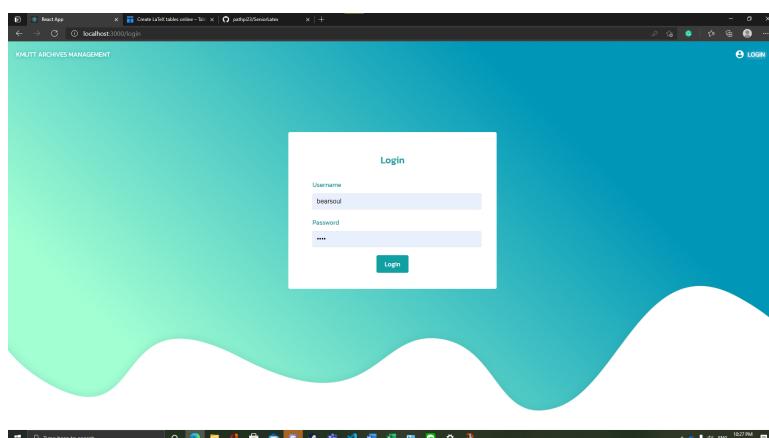
## 4.6 ผลลัพธ์ที่ได้จากการเขียนเว็บไซต์

### 4.6.1 หน้าหลัก



รูปที่ 4.1 ภาพแสดงหน้าเว็บหลัก

### 4.6.2 การเข้าสู่ระบบเว็บไซต์



รูปที่ 4.2 ภาพแสดงหน้าเข้าสู่ระบบ

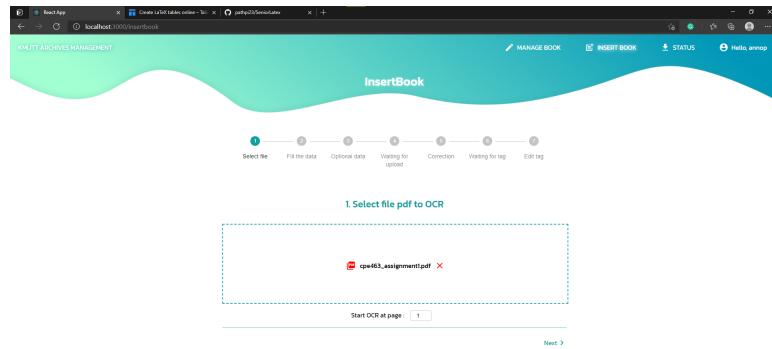
การเข้าสู่ระบบในเว็บไซต์เราได้ใช้ JSON Web Token (JWT) ในการตรวจสอบการเข้าใช้ระบบโดยที่เมื่อผู้ใช้งานเข้าสู่ระบบด้วยรหัสผู้ใช้งาน และรหัสผ่านที่ถูกต้องNode JS ก็จะคืน token ที่ถูกเข้ารหัสไว้กลับไปให้ทางเครื่องผู้ใช้งานเก็บใน local storage เพื่อที่จะเป็นการบ่งบอกสิทธิ์การใช้งาน API ที่เหลืออย่างหมดไม่ว่าจะเป็นการค้นหาข้อมูล เพิ่มข้อมูลหนังสือ แก้ไขข้อมูลหนังสือ หรือลบข้อมูลหนังสือออกจากระบบ ถ้าผู้ใช้งานไม่ได้ส่ง token มาด้วยหรือ token นั้นมีการตัดแปลงแก้ไขระบบจะทำการลบ token ภายใต้เครื่องทั้งหมดและการออกจากระบบโดยทันที

### 4.6.3 การเพิ่มหนังสือเข้าสู่ระบบฐานข้อมูล

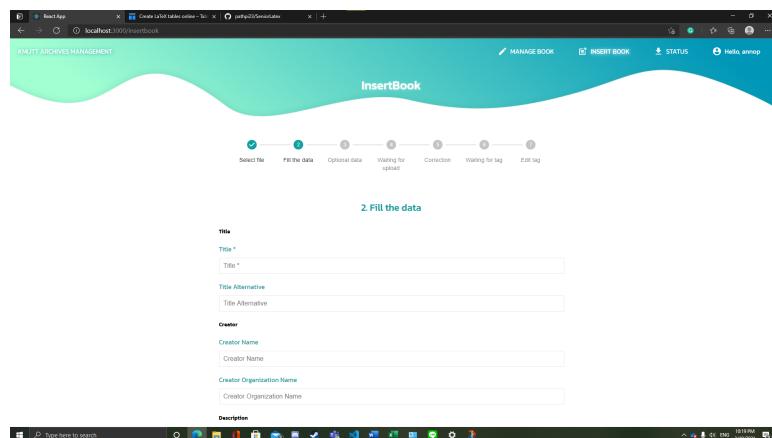
เนื่องจากการเพิ่มหนังสือเข้าสู่ระบบมีขั้นตอนจำนวนมากและใช้เวลานานจึงแบ่งการอประมวลผลเป็นส่วนของการเพิ่มข้อมูลของหนังสือ ส่วนของการแก้ไขและตรวจสอบคำกรองนำเข้าสู่ระบบ ส่วนของการตรวจสอบแก้ไขแท็ก ซึ่งผู้ใช้งานไม่จำเป็นต้องรอภายใต้หน้าเพิ่ม

หนังสือสามารถไปทำงานพังก์ชันนี้ได้ตามปกติและเมื่อเสร็จกระบวนการเหล่านี้เสร็จจะสามารถกลับมาดำเนินการเพิ่มข้อมูลต่อได้โดยการกดที่หน้า Status และกลับเข้าสู่กระบวนการเพิ่มข้อมูลหนังสือ

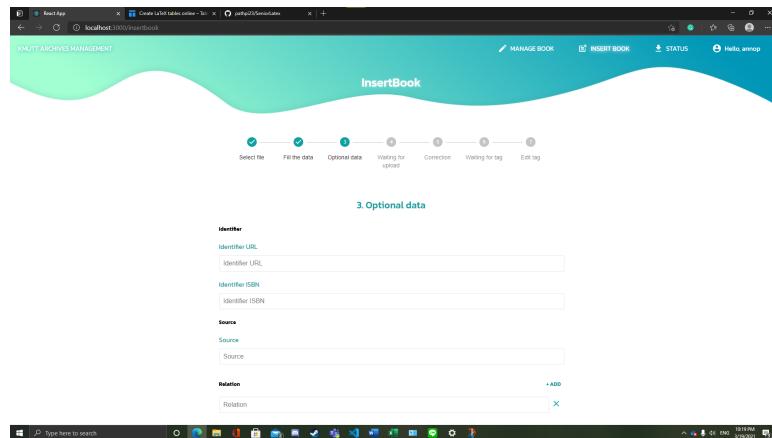
#### 4.6.3.1 เพิ่มข้อมูลของหนังสือ



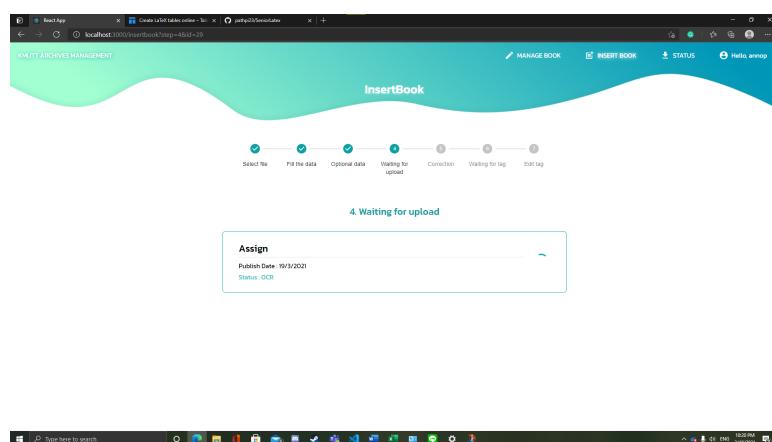
รูปที่ 4.3 ภาพแสดงขั้นตอนการเพิ่มหนังสือขั้นตอนการเพิ่มไฟล์



รูปที่ 4.4 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 1



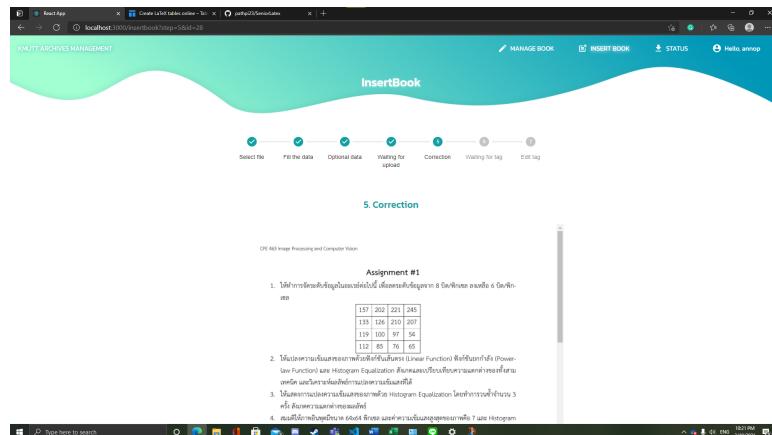
รูปที่ 4.5 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นกรอกข้อมูลขั้นที่ 2



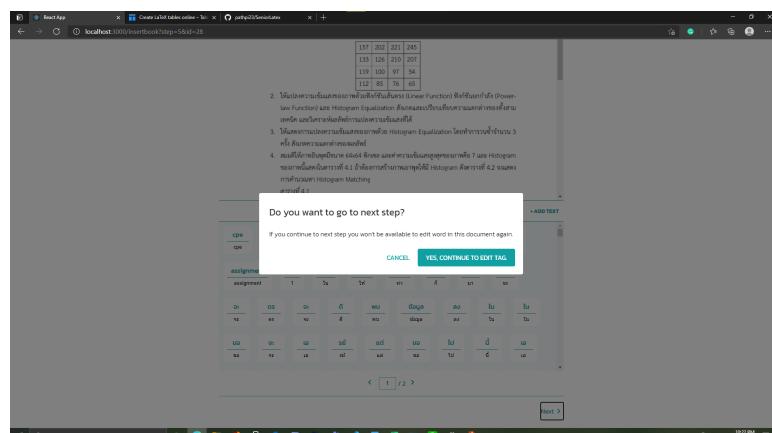
รูปที่ 4.6 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการเตรียมข้อมูล

ในส่วนนี้จะเป็นการใช้งานทำการเลือกไฟล์ PDF และกรอกข้อมูลของหนังสือโดยที่เมื่อผู้ใช้งานยืนยันข้อมูลเรียบร้อยแล้วระบบก็จะทำการเพิ่มไฟล์ PDF เพื่อนำไปทำกระบวนการเปลี่ยน PDF เป็นรูปภาพและทำการ OCR และการเตรียมข้อมูลตัวหนังสือ เพื่อทำการแปลงข้อมูลออกมาให้ผู้ใช้งาน

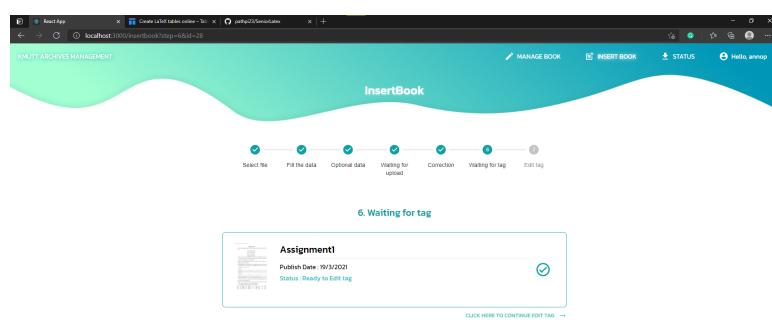
#### 4.6.3.2 การแก้ไขและตรวจสอบคำก่อนนำเข้าสู่ระบบ



รูปที่ 4.7 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการแก้ไขคำผิด



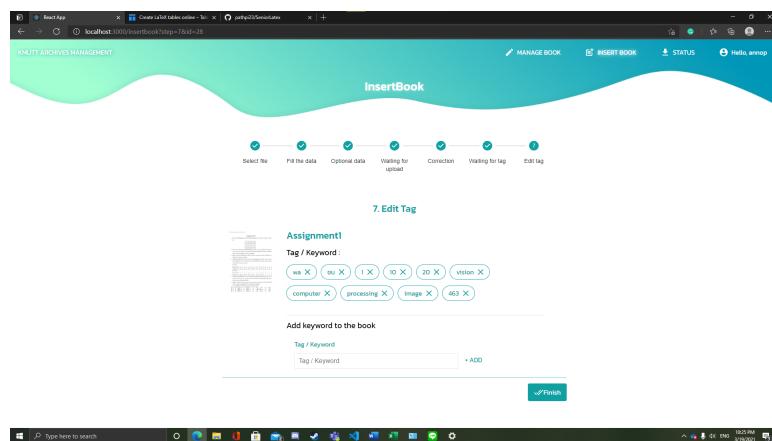
รูปที่ 4.8 ภาพแสดงหน้าต่างยืนยันการแก้คำ



รูปที่ 4.9 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการสร้างคำสำคัญ

ในส่วนนี้จะเป็นผลลัพธ์การดำเนินการของการเพิ่มข้อมูลหนังสือ จะมีคำขอของแต่ละหน้าพร้อมรูปภาพประกอบเพื่อให้ผู้ใช้งานได้ตรวจสอบ คำเพิ่มและแก้ไขคำได้อย่างอิสระก่อนจะนำคำเหล่านี้เข้าสู่ระบบและในส่วนนี้ถ้ายืนยันการแก้ไขแล้วจะไม่สามารถแก้ไขคำในหนังสือ เล่มนี้ในระบบได้อีกโดยถ้ายืนยันแล้วระบบจะทำการเพิ่มคำเหล่านี้เข้าสู่ระบบและทำการคำนวนค่า TF/IDF ของคำเหล่านี้ก่อนจะสร้างแท็ก ของหนังสือเล่มนี้ให้อัตโนมัติ

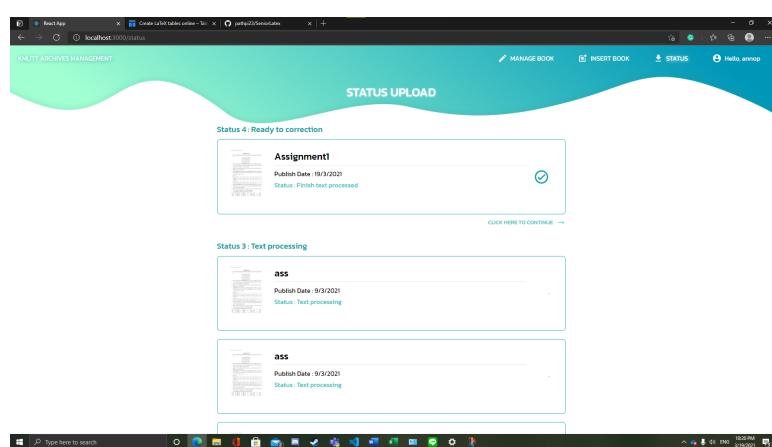
#### 4.6.3.3 การตรวจสอบแก้ไขแท็ก



รูปที่ 4.10 ภาพแสดงขั้นตอนการเพิ่มหนังสือเข้าสู่ระบบขั้นการแก้ไขคำสำคัญ

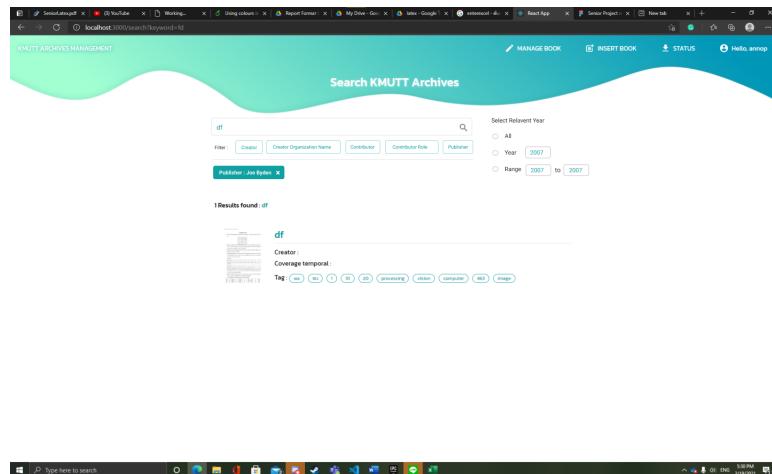
ในส่วนนี้จะเป็นผลลัพธ์ของการแก้ไขและตรวจสอบคำก่อนนำเข้าสู่ระบบโดยผู้ใช้งานได้แท็ก ที่ทางระบบทำขึ้นอัตโนมัติเพื่อให้ผู้ใช้งานได้ตรวจสอบเพิ่มลดแท็ก ก่อนจะยืนยันเพิ่มเข้าสู่ระบบ

#### 4.6.4 การแสดงสถานะการเพิ่มหนังสือ



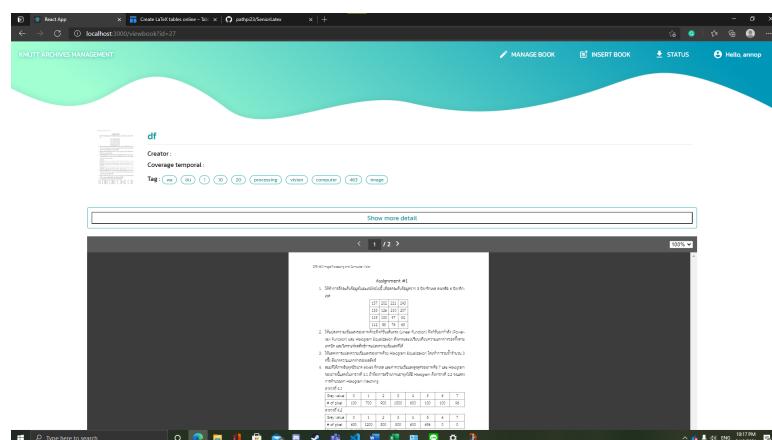
รูปที่ 4.11 ภาพแสดงสถานะของการเพิ่มข้อมูลเข้าสู่ระบบ

#### 4.6.5 การแสดงการค้นหาหนังสือ



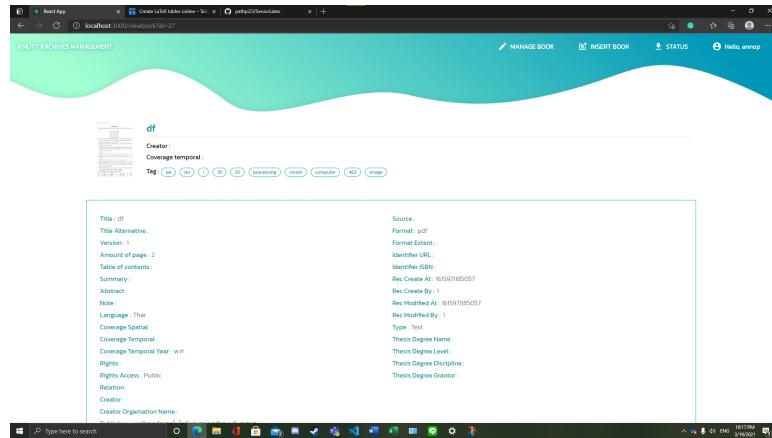
รูปที่ 4.12 ภาพแสดงหน้าการค้นหา

#### 4.6.6 การแสดงข้อมูลหนังสือ



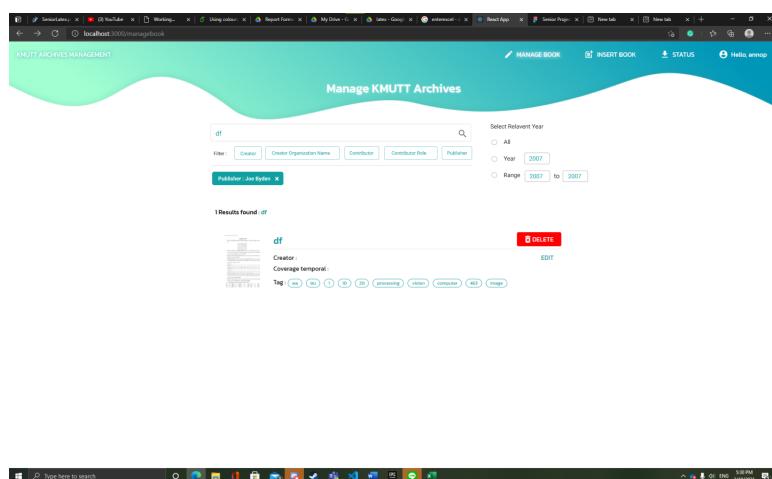
รูปที่ 4.13 ภาพแสดงหน้าแสดงหนังสือ

จะเป็นการแสดงข้อมูลของหนังสือที่อยู่ภายในระบบที่ผู้ใช้งานกรอกเข้ามาในระบบพร้อมทั้งแสดง PDF ที่ถูกอัปโหลดขึ้นมา

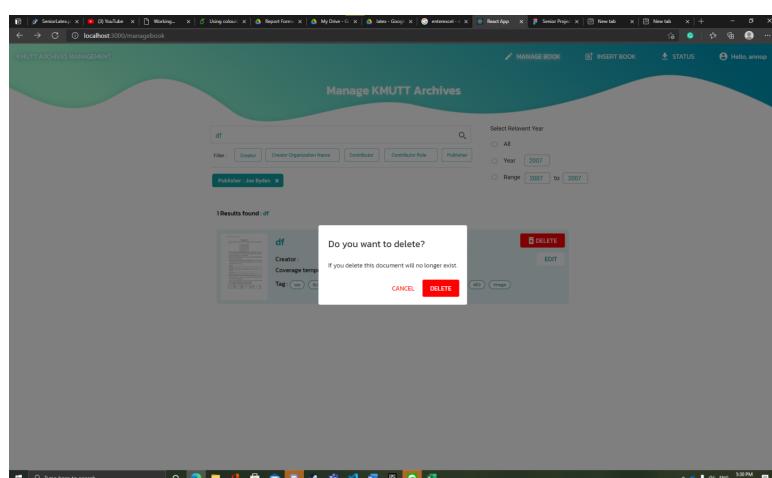


รูปที่ 4.14 ภาพแสดงข้อมูลของหนังสือ

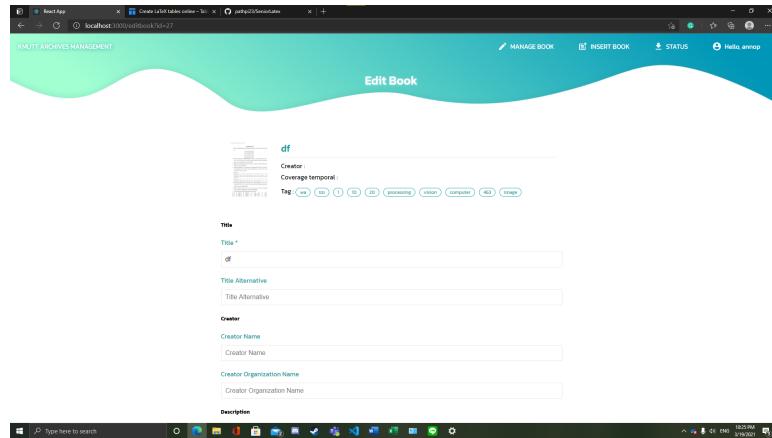
#### 4.6.7 การแสดงการแก้ไขข้อมูลของหนังสือ



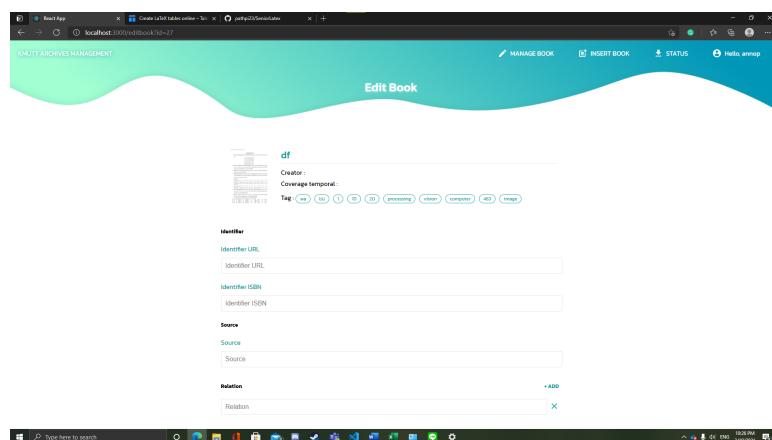
รูปที่ 4.15 ภาพแสดงหน้าการค้นหาในหน้าการจัดการหนังสือ



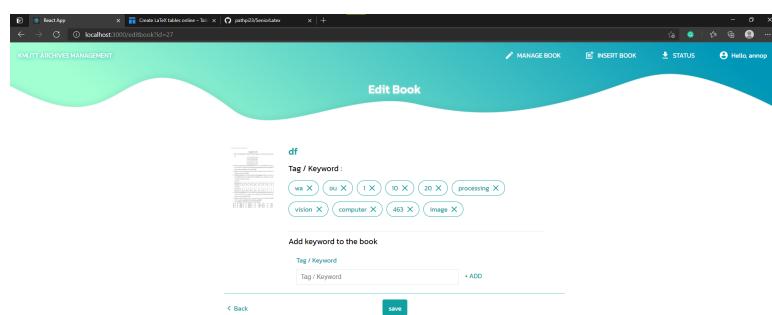
รูปที่ 4.16 ภาพแสดงหน้าการลบหนังสือ



รูปที่ 4.17 ภาพแสดงหน้าการแก้ไขข้อมูลขั้นที่ 1



รูปที่ 4.18 ภาพแสดงหน้าการแก้ไขข้อมูลขั้นที่ 2



รูปที่ 4.19 ภาพแสดงหน้าการแก้ไขคำสำคัญ

การแก้ไขข้อมูลจะแก้ได้ต่อเมื่อเพิ่มข้อมูลหนังสือเสร็จสิ้นแล้วโดยที่จะสามารถแก้ไขข้อมูลในส่วนของข้อมูลหนังสือและแท็ก ได้เมื่อมีอนกัน กับการเพิ่มหนังสือโดยเมื่อแก้ไขเสร็จสิ้นแล้วยังยังระบบจะทำการบันทึกข้อมูลใหม่ให้ทันที

## บทที่ 5 สรุปผล

### 5.1 ผลการดำเนินงาน

ในภาคการเรียนที่ 1/2563 ทางคณะผู้จัดทำได้ทำการนำเสนอหัวข้อโครงการ ศึกษาและรวบรวมข้อมูลต่างๆ เก็บ Requirement จากบรรณารักษ์ และได้ออกแบบเว็บไซต์ User Interface, โครงสร้างฐานข้อมูลและวิธีต่างๆในการสร้างเว็บไซต์ที่จะทำการแปลงรูปภาพให้อยู่ในรูปแบบดิจิทัล และทำการศึกษา เรียนรู้และออกแบบวิธีการเตรียมข้อมูลรูปภาพก่อนที่จะนำไปทำการ OCR สร้างระบบเตรียมข้อมูล ตัวหนังสือ อย่างการตัดคำ และการแก้ไขคำผิด เพื่อเตรียมข้อมูลที่ได้จากการทำ OCR ให้อยู่ในรูปแบบที่เหมาะสมสำหรับระบบการค้นหา และสร้างระบบค้นหาคำสำคัญด้วยหลักการของ TF-IDF เพื่อใช้สร้างแท็ก ของตัวหนังสือ

ณ เวลาปัจจุบันในภาคเรียนที่ 2/2563 ทางคณะผู้จัดทำได้วางแผนที่จะสร้างเว็บไซต์ จัดทำระบบการค้นหา และโมเดล Word2Vec ทำการประเมินระบบการออกแบบ User Interface การเตรียมข้อมูลรูปภาพ ระบบการแก้คำผิด และระบบการค้นหา

ตารางที่ 5.1 ตารางสรุปผลการดำเนินงาน

แผนการดำเนินการ	ยังไม่ดำเนินการ	กำลังดำเนินการ	เสร็จสิ้น
ศึกษาค้นคว้าหาปัญหาของโครงการ			
เสนอหัวข้อไปรษณีย์			
ศึกษาและหาข้อมูลเกี่ยวกับเทคโนโลยีที่ใช้ในโครงการ			
ประเมินความเป็นไปได้และกำหนดขอบเขตของโครงการ			
จัดเก็บ requirement จากกลุ่มผู้ใช้งาน			
นำเสนอโครงงานครั้งที่ 1			
ออกแบบ UX/UI			
การแปลงรูปภาพให้อยู่ในรูปแบบดิจิทัล			
นำข้อมูลที่เก็บไว้มาทำการตัดแบ่งคำภาษาไทยและทำการสร้างแท็ก โดยใช้หลักการของ TF-IDF			
จัดทำระบบการค้นหา			
จัดทำเว็บไซต์แพลตฟอร์ม			
ทดสอบระบบ			
ปรับปรุงแก้ไข			
นำเสนอไปรษณีย์	แดง		

## 5.2 ปัญหาที่พบและการแก้ไข

### 5.2.1 ปัญหาน้ำสีอ่านยาก

เนื่องจากหนังสือแต่ละเล่มมีลักษณะที่แตกต่างกันในเรื่องของสีของกระดาษและตัวอักษร ลักษณะการสแกนรูปภาพจึงทำให้การนำประโยคข้อความที่ถูกตัดออกมาทำ OCR แล้วเกิดความผิดพลาดเบื่อง

#### การแก้ไข

ทำการแก้ไขกระบวนการการเตรียมข้อมูลรูปภาพ จากการพยาຍามข้ามหน้าสีเป็นพัฒนาการลับพื้นหลังในรูปแบบใหม่

### 5.2.2 ปัญหาระบบทดลองไม่ตรง

เนื่องจากภาษาไทยมีสรุรรณยุกต์ด้านบนต่อกันสูงสุดต่อกันถึง 2 ชั้นนอกจากนั้นยังมีสรุรรณยุกต์ด้านล่างทำให้บางทีไม่สามารถแยกข้อความแต่ละบรรทัดออกจากกันได้อย่างสมบูรณ์จึงทำให้มีการหมุนที่ผิดพลาดเกิดขึ้น

#### การแก้ไข

ทำการแก้ไขกระบวนการการเตรียมข้อมูลรูปภาพ ปรับเปลี่ยนวิธีการหมุนเป็นการหา  $\arctan$  ที่จุดขอบด้านบนที่ทำให้การหมุนมีข้อผิดพลาดน้อยลง

### 5.2.3 ปัญหาระบแก้ไขคำผิด

เนื่องจากการแก้ไขคำผิดยังไม่สามารถแก้ไขคำผิดรูปแบบคำเฉพาะได้อย่างเช่นชื่อคน หรือชื่อสถานที่ หรืออาจจะคิดว่าคำเฉพาะนี้นับผิดพลาด และทำการแก้ไขให้ถูกต้องมัติทำให้คำที่ได้รับออกจากเกิดข้อผิดพลาดขึ้น

#### การแก้ไข

ให้ผู้ใช้งานได้ตรวจสอบและแก้ไขได้เองก่อจะเพิ่มข้อมูลเข้าสู่ระบบและเพิ่มข้อมูลคำเฉพาะบางส่วนลงไป

### 5.2.4 ปัญหาระบเรื่องระยะเวลาในการเพิ่มข้อมูลหนังสือ

เนื่องจากการเพิ่มข้อมูลมีขั้นตอนจำนวนมากและใช้เวลานานผู้จัดทำจึงออกแบบโครงสร้างเป็นระบบ thread เพื่อที่จะให้เซิฟเวอร์ตอบกลับไปยังผู้ใช้งานและเปิด thread ในการทำงานไม่ว่าจะเป็น OCR หรือการทำการเตรียมข้อมูลตัวหนังสือ เพื่อเพิ่มความเร็วในการทำงานแต่ก็ไม่สามารถลดเวลาการทำงานลงได้

#### การแก้ไข

ทำการแก้ไขโดยการ spawn process แทนที่จะมาเนื่องจาก python นั้นเวลาเปิด thread ยังคงใช้ core เดียวในการประมวลผลจึงต้องเปลี่ยนมาเป็นการ spawn process ที่แยกการใช้ core ของหน่วยประมวลผลทำให้สามารถลดเวลาในการเพิ่มข้อมูลหนังสือได้

### 5.3 ข้อจำกัดและข้อเสนอแนะ

1. การแก้ไขคำเฉพาะยังไม่สามารถทำได้ถึงแม้จะมีการเพิ่มคำศัพท์เฉพาะลงไปก็ตามแต่ก็ยังต้องให้มนุษย์เป็นผู้ตรวจสอบอีกรอบเพื่อความถูกต้อง
2. เนื่องจากลักษณะหนังสือแต่ละเล่มแตกต่างกันทำให้การเตรียมข้อมูลรูปภาพที่อาจจะไม่ได้ส่งผลลัพธ์ที่ดีที่สุดให้กับหนังสือทุกประเภททำให้ประสิทธิภาพในหนังสือบางเล่มน้อยกว่าหรือมากกว่าอีกเล่มได้
3. ลักษณะแสงของการสแกนหนังสือเนื่องจากไฟล์ที่ได้รับมาไม่มีการควบคุมในการสแกนหนังสือเข้ามาจึงทำให้ไฟล์มีความสว่างที่ไม่เท่ากันขนาดและการจัดวางที่ไม่เหมือนกันทำให้ประสิทธิภาพของแต่ละเล่มอาจจะไม่เท่าเดิม

## หนังสืออ้างอิง

1. Doxygen, 2020, ``OpenCV," [https://docs.opencv.org/3.4/d4/d73/tutorial\\_py\\_contours\\_begin.html](https://docs.opencv.org/3.4/d4/d73/tutorial_py_contours_begin.html), [Online; accessed 12-October-2020].
2. Fasttext, 2018, ``English word vectors," <https://fasttext.cc/docs/en/english-vectors.html>.
3. Y. Goldberg and O. Levy, 2014, ``word2vec Explained: Deriving Mikolov et al.'s," <https://arxiv.org/pdf/1402.3722v1.pdf>.
4. Google, 2020, ``Tesseract OCR," <https://opensource.google/projects/tesseract>, [Online; accessed 22-November-2020].
5. NECTEC, ``AI For Thai," <https://aiforthai.in.th/index.php#home>, [Online; accessed 22-November-2020].
6. Keiron O'Shea and Ryan Nash, 2015, ``An Introduction to Convolutional Neural Networks," [CoRR](#), vol. abs/1511.08458, 2015.
7. Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer, 2018, ``Deep contextualized word representations," [CoRR](#), vol. abs/1802.05365, 2018.
8. Ritambhara, ``Minimum edit distance of two strings," <https://www.ritambhara.in/minimum-edit-distance-of-two-strings/>, [Online; accessed 12-October-2020].
9. Xin Rong, 2014, ``word2vec Parameter Learning Explained," [CoRR](#), vol. abs/1411.2738, 2014.
10. Saixiii, 2017, ``RESTful គឺ នៅទៅ REST គឺ ការ សៀវភៅ និង ផ្តល់ព័ត៌មាន ដោយ webservice," <https://saixiii.com/what-is-restful/#:~:text=Representational%20state%20transfer%20%E0%B8%AB%E0%B8%A3%E0%B8%B7%E0%B8%AD%20REST,XML%2C%20HTML%2C%20JSON%20%E0%B9%82%E0%B8%94%E0%B8%A2%20response/>.
11. techterms, 2018, ``MVC," <https://techterms.com/definition/mvc>, [Online; accessed 10-October-2020].
12. Matt Zucker, 2016, ``Page dewarping," <https://mzucker.github.io/2016/08/15/page-dewarping.html>.