

ระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัทรีฟินิตีฟ

SOFTWARE RELEASING SYSTEM INSIDE THE REFINITIV  
CORPORATION



สหกิจศึกษาเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์  
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOFTWARE RELEASING SYSTEM INSIDE THE REFINITIV  
CORPORATION



A COOPERATIVE EDUCATION SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE  
DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2018

หัวข้อสหกิจศึกษา ระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัทรีฟิทิฟ  
Software Releasing System Inside the Refinitiv Corporation  
ชื่อนักศึกษา นางสาวสิริภรณ์ กิวัฒนา รหัสนักศึกษา 58050402  
ปริญญา วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)  
ภาควิชา วิทยาการคอมพิวเตอร์  
ปีการศึกษา 2561  
อาจารย์ที่ปรึกษา ผศ.ดร.อนันตพร หารรรษคุณาฒย

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้สหกิจศึกษานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2561

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.วิสันต์ ตั้งวงษ์เจริญ ประธานกรรมการ	
ผศ.ดร.อนันตพร หารรรษคุณาฒย กรรมการและอาจารย์ที่ปรึกษา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปะหรือเผยแพร่ไปยังผู้เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

หัวข้อสหกิจศึกษา	ระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัทรีฟินิทิว Software Releasing System Inside the Refinitiv Corporation
ชื่อนักศึกษา	นางสาวสิริภรณ์ กวีพัฒนา รหัสนักศึกษา 58050402
ปริญญา	วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2561
อาจารย์ที่ปรึกษา	ผศ.ดร.อนันตพร หารราชคุณาฒย์

### บทคัดย่อ

สหกิจศึกษานี้จัดทำขึ้นเพื่อพัฒนาและออกแบบระบบอนุมัติซอฟต์แวร์สร้างขึ้นเพื่อให้ให้บริการในซอฟต์แวร์ Eikon เนื่องจากปัจจุบันบริษัท Refinitiv มีการสร้างซอฟต์แวร์มากมายให้ลูกค้าได้ใช้งาน ซึ่งในส่วนของซอฟต์แวร์ที่สร้างขึ้นเพื่อให้บริการในซอฟต์แวร์ Eikon ก่อนการปล่อยออกสู่ตลาดไปให้บุคคลทั่วไปใช้นั้น จำเป็นต้องมีการร้องขอการปล่อย (Release) โดยใช้ระบบซอฟต์แวร์ที่ชื่อ AppRelease ซึ่งในปัจจุบันกระบวนการร้องขอและการอนุมัติมีการดำเนินการหลายขั้นตอนและใช้วิธีแบบ manual ในส่วนของการอนุมัติ ผู้อนุมัติจำเป็นต้องดำเนินการอนุมัติสองครั้ง ครั้งแรกคือการอนุมัติไปให้ผู้ใช้ภายในองค์กรใช้ ครั้งที่สองคืออนุมัติไปให้บุคคลทั่วไปใช้ ซึ่งหลังการอนุมัติครั้งแรกระบบจะดำเนินการนานหลายวัน จึงจะแสดงสถานะให้สามารถอนุมัติรอบที่สองได้ เนื่องจากการอนุมัติรอบที่สองจำเป็นต้องใช้ระยะเวลาหลายวันจึงจะอนุมัติได้ ทำให้ผู้อนุมัติเกิดการหลังสัการอนุมัติได้ หากมีการกำหนดวันปล่อยแอปพลิเคชันดังกล่าวไว้แน่นอน ก็จะทำให้การปล่อยแอปพลิเคชันดังกล่าวล่าช้าได้ จึงเป็นเหตุให้มีการสร้างระบบใหม่ที่ช่วยอำนวยความสะดวกในการปล่อยซอฟต์แวร์ที่สร้างขึ้น ให้ผู้มีสิทธิ์อนุมัติสามารถอนุมัติซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรได้แบบอัตโนมัติโดยไม่จำเป็นต้องเสียเวลาดำเนินการหลายขั้นตอน โดยระบบจะดำเนินการอนุมัติครั้งที่หนึ่งและครั้งที่สองให้อย่างอัตโนมัติโดยทำการกดปุ่มอนุมัติเพียงปุ่มเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Software Releasing System Inside the Refinitiv Corporation
Student	Miss Siriporn Kiwatthana Student ID 58050402
Degree	Bachelor of Science (Computer Science)
Department	Computer Science
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2018
Advisor	Asst.Prof.Dr. Anantaporn Hanskunatai

### Abstract

The objective of this cooperative education is to create the Software Releasing System Inside the Refinitiv Corporation. Refinitiv Software (Thailand) Limited has many software products available to customers. In respect of Eikon software and applications publication, it requires a release request by using the software system called AppReleaseJ. Currently, there are many steps and methods used by manual in the request and approval release process. In approving, there are two times. The first time is released approving for the users within the corporation and the second time is released approving for the users on the Eikon software. When an approver approved release, it is necessary to wait in several days for the second approve. As a result, the approver may forget in the second approval. If the software has a fixed release date, it will cause delayed releasing. This reason led to the development of a new system that facilitates the releasing software and applications. Approver can automatically approve software and applications within an organization without having to spend a lot of time for processing them. The system will automatically approve the first and the second time by only click on one button.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

การจัดทำสหกิจศึกษาหัวข้อ Software Releasing System Inside the Refinitiv Corporation สำเร็จลุล่วงไปได้ด้วยดีเนื่องจากผู้จัดทำได้รับความช่วยเหลือจากบุคคลผู้ที่มีพระคุณหลายท่านดังนี้

ขอขอบพระคุณ บิดา มารดา และสมาชิกในครอบครัวทุกคน ผู้ซึ่งอยู่เบื้องหลังคอยอบรมสั่งสอน โดยให้กำลังใจและดูแลตลอดมา ในการจัดทำสหกิจศึกษานี้

ขอขอบพระคุณ อาจารย์วิสันต์ ตั้งวงษ์เจริญ และ ผศ.ดร.อนันตพร ทรราชคุณาฒย์ อาจารย์ที่ปรึกษาในการทำหัวข้อสหกิจศึกษานี้ ที่ได้ให้คำปรึกษาและคำแนะนำในการแก้ไขปัญหาโดยตลอด รวมถึงการตรวจสอบและแก้ไขการเขียนหัวข้อสหกิจศึกษานี้อย่างละเอียด จนสามารถดำเนินการสำเร็จลุล่วงได้อย่างดี

ขอขอบพระคุณ นายอดิภัทร ลาภรัตนากุล นายวีระยุทธ ชัยศิริถาวกุล และบริษัท Refinitiv Thailand ที่คอยให้ความช่วยเหลือ ให้คำแนะนำและให้โอกาสมาทำงานจริง เรียนรู้การทำงานในบริษัท จนสามารถดำเนินการสำเร็จลุล่วงได้อย่างดี

ขอขอบคุณคณาจารย์ภาควิชาการคอมพิวเตอร์ทุกท่านที่ได้ถ่ายทอดความรู้และอบรมสั่งสอน ทั้งในภาคทฤษฎีและภาคปฏิบัติ ให้แก่ทางคณะผู้พัฒนามาตลอดระยะเวลา 4 ปี ให้สำเร็จลุล่วงไปได้

สิริภรณ์ กิวัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ .....	ข
กิตติกรรมประกาศ .....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูปภาพ .....	ช
<b>บทที่ 1 บทนำ.....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการสหกิจศึกษา.....	1
1.3 ขอบเขตของสหกิจศึกษา .....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ .....	2
1.5 เครื่องมือและภาษาที่ใช้ในโครงการสหกิจศึกษา.....	2
<b>บทที่ 2 เทคโนโลยีและระบบงานที่เกี่ยวข้อง .....</b>	<b>3</b>
2.1 ลำดับขั้นตอนการปล่อยแอปพลิเคชันของ AppReleaseJ ในปัจจุบัน.....	3
2.2 ปัญหาที่พบในปัจจุบัน.....	6
2.3 เทคโนโลยีที่ใช้ในการพัฒนาระบบ .....	6
2.3.1 Application Programming Interface (API).....	6
2.3.2 REST API (Representational State Transfer) .....	7
2.3.3 ภาษา HTML .....	8
2.3.4 MongoDB.....	8
2.3.5 Postman.....	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ(ต่อ)

หน้า

2.3.6 Sysinternals Process Explorer.....	9
2.3.7 Angular 6 Frontend Frameworks .....	10
2.3.8 Visual Studio Code .....	10
2.3.9 Jenkins .....	11
2.3.10 Docker .....	11
2.3.11 Continuous Integration (CI).....	13
<b>บทที่ 3 ขั้นตอนการพัฒนาาระบบ.....</b>	<b>14</b>
3.1 โครงสร้างโดยรวมของระบบ.....	14
3.2 การวิเคราะห์ระบบ .....	16
3.2.1 Functional Requirement.....	16
3.2.2 ข้อมูลจากระบบอื่นที่นำมาใช้ในระบบ.....	19
3.2.3 Use Case Diagram.....	20
3.2.4 Activity Diagram.....	27
3.2.5 Sequence Diagram.....	37
3.2.6 การออกแบบฐานข้อมูล.....	49
<b>บทที่ 4 ผลการดำเนินงาน .....</b>	<b>56</b>
4.1 ผลลัพธ์การพัฒนาเว็บเอพีไอ (Web API Service) ของระบบ.....	56
4.2 ผลลัพธ์การพัฒนาเว็บแอปพลิเคชันของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายใน องค์กรของบริษัท Refinitiv.....	57
4.3 ผลลัพธ์การทดสอบระบบแอปพลิเคชันของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายใน องค์กรของบริษัท Refinitiv.....	74
<b>บทที่ 5 สรุปผลการดำเนินงาน และข้อเสนอแนะ.....</b>	<b>82</b>
5.1 สรุปผลการดำเนินงาน .....	82
5.2 ข้อเสนอแนะ.....	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่าเอกสารอ้างอิงอื่น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างถึงถึงเจ้าของเอกสารทุกครั้งที่มีการนำ: 83

## สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 3.1 อธิบายการเปลี่ยน state ของแอปพลิเคชันในระบบ.....	18
ตารางที่ 3.2 Use Case Documentation ของฟังก์ชัน Create release.....	22
ตารางที่ 3.3 Use Case Documentation ของฟังก์ชัน View state release.....	22
ตารางที่ 3.4 Use Case Documentation ของฟังก์ชัน View detail release.....	23
ตารางที่ 3.5 Use Case Documentation ของฟังก์ชัน Signed off release.....	23
ตารางที่ 3.6 Use Case Documentation ของฟังก์ชัน Approved release.....	24
ตารางที่ 3.7 Use Case Documentation ของฟังก์ชัน Withdrawn release.....	24
ตารางที่ 3.8 Use Case Documentation ของฟังก์ชัน AppName Search.....	25
ตารางที่ 3.9 Use Case Documentation ของฟังก์ชัน Sort by state.....	25
ตารางที่ 3.10 Use Case Documentation ของฟังก์ชัน Login.....	26
ตารางที่ 4.1 ผลการทดสอบฟังก์ชันลงชื่อเข้าใช้งานระบบ.....	75
ตารางที่ 4.2 ผลการทดสอบฟังก์ชันสร้างรายชื่อแอปพลิเคชันที่ต้องการร้องขอ release.....	76
ตารางที่ 4.3 ผลการทดสอบฟังก์ชัน Signed off แอปพลิเคชันที่มีการร้องขอ release.....	77
ตารางที่ 4.4 ผลการทดสอบฟังก์ชันอนุมัติแอปพลิเคชันที่ต้องการ release ไป internal.....	77
ตารางที่ 4.5 ผลการทดสอบฟังก์ชันยกเลิกการ release แอปพลิเคชัน.....	78
ตารางที่ 4.6 ผลการทดสอบฟังก์ชัน View detail release.....	78
ตารางที่ 4.7 ผลการทดสอบฟังก์ชันค้นหารายชื่อแอปพลิเคชันที่มีการ create release.....	79
ตารางที่ 4.8 ผลการทดสอบฟังก์ชัน Sort by state.....	80
ตารางที่ 4.9 การประเมินผลการสร้างระบบ Software Releasing System Inside the Refinitiv Corporation.....	81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปร่างภาพ

รูปที่	หน้า
รูปที่ 2.1 ขั้นตอนการดำเนินการร้องขอให้อนุมัติแอปพลิเคชัน.....	4
รูปที่ 2.2 แผนภาพแสดงลำดับขั้นตอนการปล่อยแอปพลิเคชันของ AppReleaseJ ในปัจจุบัน.....	5
รูปที่ 2.3 Application Programming Interface (API).....	6
รูปที่ 2.4 REST API Architecture.....	7
รูปที่ 2.5 การเก็บค่าเป็น key และ value ของ JSON.....	8
รูปที่ 2.6 สถาปัตยกรรมการทำงานของ Angular Framework.....	10
รูปที่ 2.7 workflow การทำงานของ Jenkins.....	11
รูปที่ 2.8 การเปรียบเทียบ container กับ virtual machines.....	12
รูปที่ 2.9 แนวการพัฒนาระบบ Continuous Integration (CI).....	13
รูปที่ 3.1 แผนภาพแสดงโครงสร้างโดยรวมของระบบ.....	14
รูปที่ 3.2 workflows การเปลี่ยน state ของระบบเว็บแอปพลิเคชัน.....	17
รูปที่ 3.3 Use Case Diagram ของระบบ.....	20
รูปที่ 3.4 Activity Diagram ของหน้าเว็บ Create Release.....	27
รูปที่ 3.5 Activity Diagram ของหน้าเว็บ View State Release.....	28
รูปที่ 3.6 Activity Diagram ของหน้าเว็บ View Detail Release.....	29
รูปที่ 3.7 Activity Diagram ของหน้าเว็บ Signed off การร้องขอ Release.....	30
รูปที่ 3.8 Activity Diagram ของหน้าเว็บ Approved การร้องขอ Release.....	32
รูปที่ 3.9 Activity Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release.....	33
รูปที่ 3.10 Activity Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการRelease.....	34
รูปที่ 3.11 Activity Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะ ของแอปพลิเคชัน.....	35
รูปที่ 3.12 Activity Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ.....	36
รูปที่ 3.13 Sequence Diagram ของหน้าเว็บ Create Release.....	37
รูปที่ 3.14 Sequence Diagram ของหน้าเว็บ View State Release.....	38
รูปที่ 3.15 Sequence Diagram ของหน้าเว็บ View Detail Release.....	39
รูปที่ 3.16 Sequence Diagram ของหน้าเว็บ Signed off การร้องขอ Release.....	40
รูปที่ 3.17 Sequence Diagram ของหน้าเว็บ Approved การร้องขอ Release.....	42
รูปที่ 3.18 Sequence Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release.....	44

## สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
รูปที่ 3.19 Sequence Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการ Release.	46
รูปที่ 3.20 Sequence Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน.....	47
รูปที่ 3.21 Sequence Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ.....	48
รูปที่ 3.22 โครงสร้างการเก็บข้อมูล application หลังจากการ create release .....	49
รูปที่ 3.23 โครงสร้างการเก็บข้อมูล application หลังจากการกด signed off.....	50
รูปที่ 3.24 โครงสร้างการเก็บข้อมูล application หลังจากการกด approved.....	51
รูปที่ 3.25 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น internal.....	52
รูปที่ 3.26 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น wait for Everyone	53
รูปที่ 3.27 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น Everyone.....	54
รูปที่ 3.28 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น Withdrawn .....	55
รูปที่ 4.1 หน้าแรกของเว็บแอปพลิเคชัน .....	57
รูปที่ 4.2 หน้าเว็บแสดง Login .....	58
รูปที่ 4.3 หน้าเว็บแสดงการกรอก username และ password .....	59
รูปที่ 4.4 หน้าเว็บแสดงผลการ Login ผิดพลาด.....	59
รูปที่ 4.5 หน้าเว็บแสดงผลหน้าหลักหลังจากการ login สำเร็จ .....	60
รูปที่ 4.6 หน้าเว็บแสดงผลการเก็บ cookie หลังจากการ login สำเร็จ.....	60
รูปที่ 4.7 หน้าเว็บของการ create release.....	61
รูปที่ 4.8 หน้าเว็บแสดงการกรอกข้อมูลแอปพลิเคชันที่ต้องการ release .....	61
รูปที่ 4.9 หน้าเว็บแสดงผลการ create release สำเร็จ .....	62
รูปที่ 4.10 หน้าเว็บรายการ release ทั้งหมด หลังการ create release สำเร็จ.....	62
รูปที่ 4.11 หน้าเว็บแสดงการเลือกที่ชื่อของแอปพลิเคชัน .....	63
รูปที่ 4.12 หน้าเว็บแสดงรายละเอียดของแอปพลิเคชัน releaseA.....	63
รูปที่ 4.13 หน้าเว็บแสดงการค้นหารายชื่อแอปพลิเคชันที่มีการ create release.....	64
รูปที่ 4.14 หน้าเว็บแสดงรายการ release ทั้งหมด หลังการ sort by state.....	64
รูปที่ 4.15 หน้าเว็บแสดงปุ่มสำหรับกด signed off แอปพลิเคชันที่มีการ release .....	65
รูปที่ 4.16 หน้าเว็บแสดงผลหลังจากกดปุ่ม signed off .....	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
รูปที่ 4.17 หน้าเว็บแสดง response จากAppReleaseJ หลังจากการ addNewRule สำเร็จ.....	66
รูปที่ 4.18 หน้าเว็บแสดงผลหลังการ signed off สำเร็จ .....	66
รูปที่ 4.19 หน้าเว็บแสดงปุ่มสำหรับกด approved แอปพลิเคชันที่มีการ release.....	67
รูปที่ 4.20 หน้าเว็บแสดงผลหลังจากกดปุ่ม approved .....	68
รูปที่ 4.21 หน้าเว็บแสดงข้อความ Update State หลังจากการคลิกปุ่ม ok.....	68
รูปที่ 4.22 หน้าเว็บแสดงการร้องขอ UpdateWorkflowTicket และ HistoryChangeState .....	69
รูปที่ 4.23 หน้าเว็บแสดงผลหลังการ approved สำเร็จ .....	69
รูปที่ 4.24 หน้าเว็บแสดงปุ่มสำหรับกด withdrawn แอปพลิเคชันที่มีการ release.....	70
รูปที่ 4.25 หน้าเว็บแสดงผลหลังจากกดปุ่ม withdrawn.....	70
รูปที่ 4.26 หน้าเว็บแสดงข้อความ Update State หลังจากการคลิกปุ่ม ok.....	71
รูปที่ 4.27 หน้าเว็บแสดงผลหลังการ withdrawn สำเร็จ.....	71
รูปที่ 4.28 หน้าเว็บแสดงสถานะ Internal ของแอปพลิเคชัน AdipatWebDemo.....	72
รูปที่ 4.29 หน้าเว็บแสดงสถานะ wait for Everyone ของแอปพลิเคชัน AdipatWebDemo.....	72
รูปที่ 4.30 หน้าเว็บแสดงสถานะ Everyone ของแอปพลิเคชัน AdipatWebDemo.....	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันบริษัท Refinitiv มี software products มากมายซึ่ง Eikon เป็น software สำคัญที่ให้บริการสำหรับผู้เชี่ยวชาญด้านการเงิน ในการแสดงผลและวิเคราะห์ข้อมูลทางการเงิน ช่วยให้สามารถเข้าถึงข้อมูลตลาดแบบเรียลไทม์ เช่น ข้อมูลข่าว ข้อมูลพื้นฐาน เป็นต้น มีเครื่องมือช่วยในการวิเคราะห์การซื้อขาย ภายใน Eikon มีแอปพลิเคชันมากมายให้บริการ เนื่องจาก Eikon เป็น software ที่เกี่ยวข้องทางด้านการเงิน แอปพลิเคชันต่างๆภายใน Eikon จึงจำเป็นต้องมีความเสถียรภาพสูง นักพัฒนาภายในองค์กรมากกว่า 400 คนที่พัฒนาแอปพลิเคชันภายใน Eikon ใดๆขึ้นเมื่อพัฒนาสำเร็จ จำเป็นต้องทำการร้องขอ Release แอปพลิเคชันดังกล่าว โดยปัจจุบันมีการดำเนินการร้องขอผ่านระบบเว็บ AppReleaseJ เมื่อมีการขอร้อง Release แอปพลิเคชันใดๆขึ้น ผู้อนุมัติหรือนักพัฒนา/ผู้บริหารระดับสูงที่มีสิทธิ์อนุมัติจะทำการอนุมัติการร้องขอดังกล่าว โดยการอนุมัติรอบแรกจะเป็นการอนุมัติไปให้บุคคลภายในองค์กรใช้ และการอนุมัติรอบที่สองเป็นการอนุมัติไปให้บุคคลทั่วไปใช้งานได้(ปล่อยออกสู่ตลาด) ซึ่งปัจจุบันการร้องขอและการอนุมัติมีขั้นตอนแบบ manual หลายขั้นตอน และระยะเวลาในการอนุมัติแต่ละรอบใช้เวลาเวลานานไม่มีความแน่นอน ทำให้ผู้อนุมัติเกิดการหลงลืมการอนุมัติไปได้ จึงเป็นเหตุให้มีการสร้างระบบใหม่ที่จะช่วยอำนวยความสะดวกในการปล่อยซอฟต์แวร์ที่สร้างขึ้น ให้ผู้มีสิทธิ์อนุมัติสามารถอนุมัติซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรได้แบบอัตโนมัติโดยไม่จำเป็นต้องเสียเวลาดำเนินการหลายขั้นตอน

### 1.2 วัตถุประสงค์ของโครงการสหกิจศึกษา

- 1) พัฒนาระบบใหม่เพื่อลดระยะเวลา รวมถึงลดขั้นตอนในการร้องขอและการอนุมัติซอฟต์แวร์ที่ต้องการปล่อยสู่ตลาด ให้ผู้ร้องขอและผู้อนุมัติสามารถร้องขอและอนุมัติซอฟต์แวร์ดังกล่าวได้ โดยดำเนินการกดปุ่มเพียงปุ่มเดียว
- 2) เพื่อให้ให้นักพัฒนาภายในองค์กรได้รับความสะดวกมากขึ้น ปรับเปลี่ยนขั้นตอนวิธีการร้องขอและอนุมัติแบบ manual ให้ดำเนินการแบบอัตโนมัติ สามารถมองเห็นสถานะทั้งหมดของแอปพลิเคชันที่กำลังพัฒนาได้อย่างง่ายขึ้น มีการแสดงสถานะการปล่อยแอปพลิเคชันแบบเป็น

เอกสารนี้เป็นเอกสารลับ  
เอกสารนี้เป็นเอกสารลับชั้นสูง ควรใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1.3 ขอบเขตของสหกิจศึกษา

- 1) พัฒนาระบบใหม่เพื่อใช้ในการอนุมัติซอฟต์แวร์ที่ต้องการปล่อยออกสู่ตลาดได้อย่างอัตโนมัติ มีการเก็บข้อมูลสถานะไว้ในฐานข้อมูลเพื่อใช้ในการแสดงสถานะบนระบบ
- 2) พัฒนาระบบในส่วนติดต่อผู้ใช้งาน (front-end) เพื่อใช้งานระบบในการร้องขอและอนุมัติ
- 3) พัฒนาระบบในส่วนของ Service (back-end) โดยใช้ API เป็นตัวกลางในการสื่อสารและแลกเปลี่ยนข้อมูลกันระหว่าง client และ service
- 4) ระบบที่พัฒนาขึ้นใหม่นี้สามารถใช้ได้เฉพาะภายในองค์กรเท่านั้น ในส่วนการเขียนโปรแกรมจะไม่นำระบบเดิมมาแก้ไข แต่จะทำการสร้างระบบใหม่เพื่อช่วยเหลือระบบเดิม

### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ช่วยลดระยะเวลาในการปล่อยแอปพลิเคชันที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv โดยหลังจากทำการเลือกอนุมัติแอปพลิเคชัน ระบบจะดำเนินการอนุมัติโดยอัตโนมัติ ทำให้ผู้ใช้งานไม่จำเป็นต้องใช้วิธีการแบบ manual ในการทำขั้นตอนการอนุมัติแอปพลิเคชันแบบระบบเดิม
- 2) เพื่อให้ผู้ใช้งานสามารถมองเห็นสถานะทั้งหมดของแอปพลิเคชันที่กำลังพัฒนาได้อย่างง่ายขึ้น มีการแสดงสถานะการปล่อยแอปพลิเคชัน (Releasing) แบบเป็นลำดับขั้น

### 1.5 เครื่องมือและภาษาที่ใช้ในโครงการสหกิจศึกษา

- 1) ฮาร์ดแวร์ (Hardware) ที่ใช้ในการพัฒนาระบบ ประกอบด้วย
  - เครื่องคอมพิวเตอร์โน้ตบุ๊กยี่ห้อ Dell Windows 7 ของบริษัท Refinitiv
  - มีหน่วยประมวลผล (CPU) Intel(R) Core(TM) i7-4810MQ CPU @ 2.80GHz 2.79 GHz
  - ฮาร์ดดิสก์ (Hard disk) ขนาดความจุ 238.47 กิกะไบต์
  - หน่วยความจำหลักขนาด 8 กิกะไบต์
- 2) ซอฟต์แวร์ (Software) ที่ใช้ในการพัฒนาระบบ ประกอบด้วย
  - โปรแกรม Microsoft Visual Studio Code ใช้ในการพัฒนาระบบด้วยภาษาจาวาสคริปต์ (JavaScript Programming Language)
  - โปรแกรม MongoDB ใช้ในการจัดการฐานข้อมูลของระบบ
  - โปรแกรม Postman ใช้ในการจัดการกับ API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เทคโนโลยีและระบบงานที่เกี่ยวข้อง

ในบทนี้เป็นการกล่าวถึงเทคโนโลยีและระบบงานที่เกี่ยวข้องของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv โดยจะกล่าวถึงลำดับขั้นตอนการปล่อยแอปพลิเคชันของ AppReleaseJ ในปัจจุบัน ปัญหาที่พบ และเทคโนโลยีที่เกี่ยวข้องที่ใช้ในระบบ

### 2.1 ลำดับขั้นตอนการปล่อยแอปพลิเคชันของ AppReleaseJ ในปัจจุบัน

AppReleaseJ เป็นเว็บแอปพลิเคชันที่สำคัญของบริษัท Refinitiv เป็นเสมือนเครื่องมือในการตรวจสอบและอนุมัติแอปพลิเคชันต่างๆก่อนปล่อยไปสู่ตลาด เมื่อนักพัฒนาภายในองค์กรพัฒนาแอปพลิเคชันสำหรับ Eikon ใดๆขึ้น หากต้องการปล่อยแอปพลิเคชันดังกล่าวให้บุคคลทั่วไปใช้จำเป็นต้องใช้ AppReleaseJ ในการร้องขอให้ผู้อนุมัติทำการอนุมัติ ผู้อนุมัติจำเป็นต้องทำการอนุมัติสองรอบ รอบแรกเป็นการอนุมัติเพื่อให้แอปพลิเคชันถูก Release ไปให้บุคคลภายในองค์กรใช้ (TR Internal) และรอบที่สองเป็นการอนุมัติเพื่อให้แอปพลิเคชันถูก Release ไปให้ผู้ใช้ทั่วไปที่ใช้ซอฟต์แวร์ Eikon ใช้ (Everyone) โดยจะมี data center เป็นตัวกำหนดว่าจะ release ไปภูมิภาคใดบ้าง โดยกระบวนการร้องขอและการอนุมัติด้วยระบบเว็บแอปพลิเคชัน AppReleaseJ มีขั้นตอนดังนี้

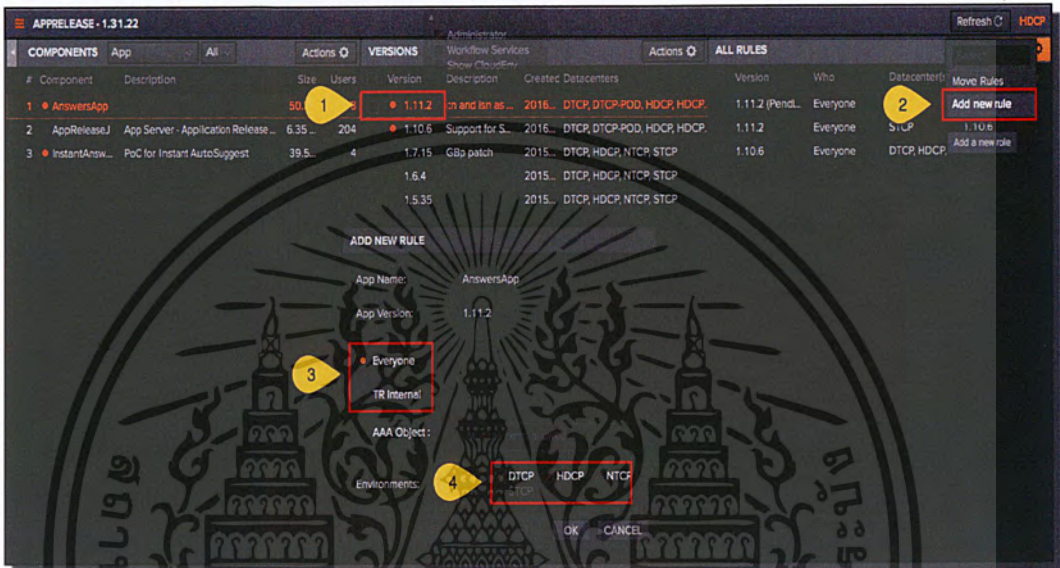
- 1) เมื่อผู้พัฒนาแอปพลิเคชันพร้อมที่จะทำขั้นตอน Release แล้ว (สิทธิ์ของผู้ใช้งานแอปพลิเคชันเริ่มต้นคือผู้พัฒนาแอปพลิเคชัน) ผู้พัฒนาแอปพลิเคชันจะต้องทำการ Release แอปพลิเคชันที่สร้างให้ QA ทำการตรวจสอบ และรอ QA อนุมัติ โดยผู้พัฒนาสามารถทำการเขียนอธิบายว่าได้ทำการพัฒนาในส่วนใดไปแล้วบ้าง (Release Note) ซึ่งข้อความจะถูกส่งผ่านทางอีเมล
- 2) ในระหว่างนั้นถ้ามีปัญหา QA จะทำการแจ้งปัญหาที่เกิดขึ้นให้ผู้พัฒนาทราบ ผู้พัฒนาจำเป็นต้องสนทนาสื่อสารกับ QA เพื่อแก้ปัญหาดังกล่าวผ่านทางอีเมล
- 3) หลังจาก QA ทำการอนุมัติ ผู้พัฒนาจะต้องทำการร้องขอให้เปลี่ยนสถานะของผู้มีสิทธิ์ใช้แอปพลิเคชันจากตัวผู้พัฒนาเองเป็นบุคคลภายในองค์กร เพื่อให้บุคคลภายในองค์กรได้ทดลองใช้งานแอปพลิเคชันที่สร้างขึ้น โดยผู้ที่มีสิทธิ์ในการอนุมัติจะทำการอนุมัติให้ ซึ่งการดำเนินการร้องขอให้อนุมัติมีขั้นตอนดังนี้

เอกสารนี้เป็3.1 เลือก version ที่ต้องการร้องขอให้เปลี่ยนสิทธิ์ของผู้ใช้งาน อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณี3.2 คลิกที่ Action ตรงมุมด้านขวา จากนั้นเลือก "Add new rule" เอกสารทุกครั้งที่มีการนำไปใช้

3.3 เลือกสิทธิ์ผู้ใช้งานที่ต้องการ โดยที่ Everyone คือ ผู้ใช้งานทั่วไป , TR Internal คือ บุคคลภายในองค์กร

3.4 เลือกภูมิภาคที่ต้องการจะ Release แอปพลิเคชัน จากนั้นเลือกปุ่ม OK

โดยในขั้นตอนที่ 3 จะมีการดำเนินการดังรูปที่ 2.1



รูปที่ 2.1 ขั้นตอนการดำเนินการร้องขอให้อนุมัติแอปพลิเคชัน

4) เมื่อนักพัฒนาทำการร้องขอ ผู้อนุมัติจะทำการตรวจสอบและทำการอนุมัติ ซึ่งการดำเนินการอนุมัติมีขั้นตอนดังนี้

4.1 ผู้อนุมัติลงชื่อเข้าใช้ระบบ AppReleaseJ

4.2 เลือกแอปพลิเคชันที่ต้องการอนุมัติ

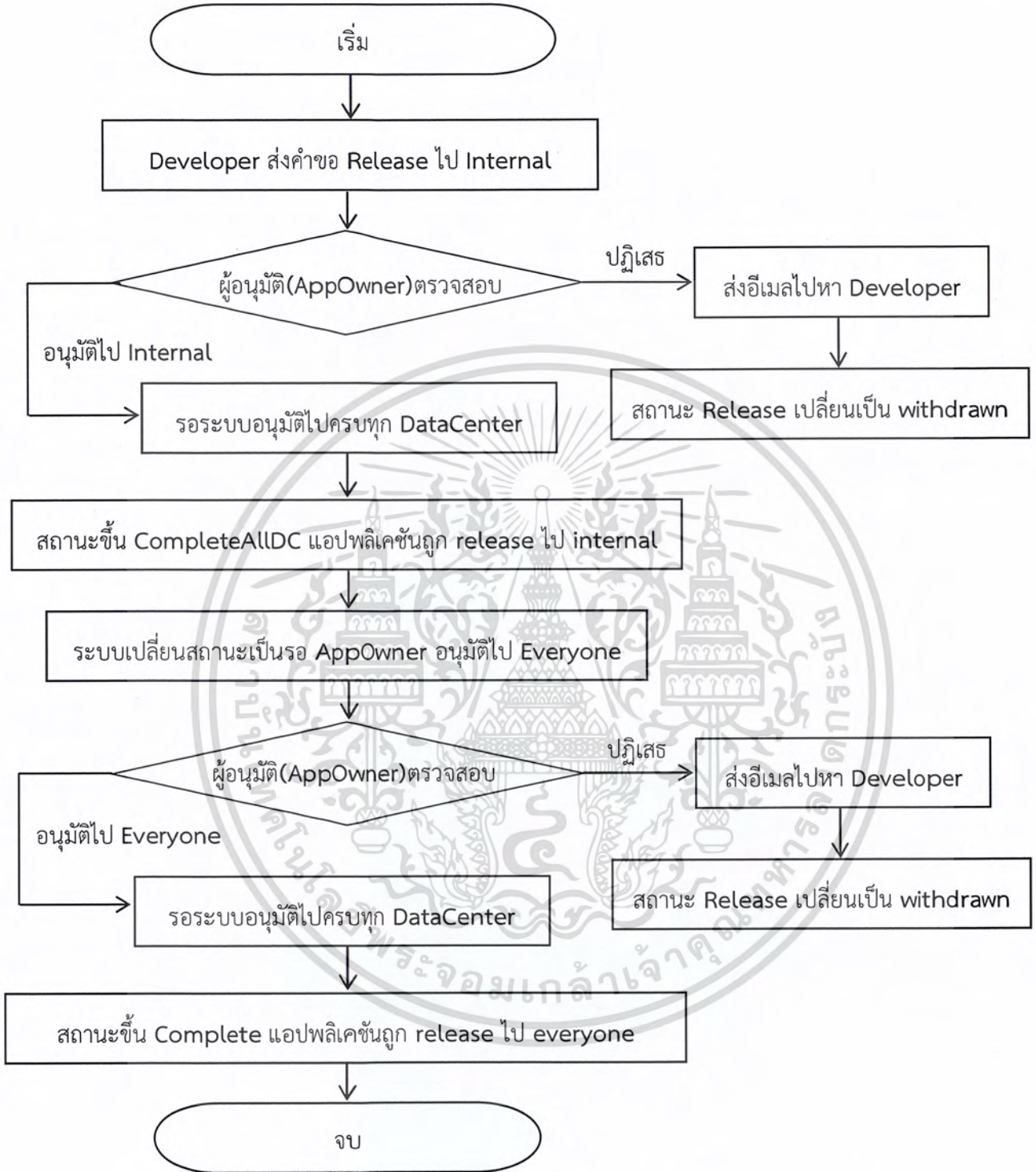
4.3 ตรวจสอบ version ที่มีการร้องขอให้อนุมัติ ที่ All Rules

4.4 คลิกขวาที่ version ดังกล่าว เพื่อ navigate ไปยังหน้าอนุมัติ

4.5 ทำการอนุมัติแอปพลิเคชันดังกล่าว

5) หลังจากแอปพลิเคชันดังกล่าว release ไปให้บุคคลภายในองค์กรใช้แล้ว ระบบจะขึ้นสถานะให้ผู้อนุมัติทำการอนุมัติรอบที่สอง เพื่อเปลี่ยนสถานะของผู้มีสิทธิ์ใช้แอปพลิเคชันจากบุคคลภายในองค์กรเป็นผู้ใช้ทั่วไปที่ใช้บริการ Eikon เพื่อให้ผู้ใช้ทั่วไปได้ใช้แอปพลิเคชันดังกล่าว ผู้ที่มีสิทธิ์

อนุมัติจะทำการตรวจสอบและทำการอนุมัติให้ตามกระบวนการ (4.1-4.5) เมื่อผู้อนุมัติทำการอนุมัติ ระบบจะทำการอัปเดตสถานะ ซึ่งจะใช้เวลาประมาณ 3 วัน แอปพลิเคชันดังกล่าวจึงจะถูก release ไปให้ผู้ใช้ทั่วไปใน Eikon ใช้บริการได้



รูปที่ 2.2 แผนภาพแสดงลำดับขั้นตอนการปล่อยแอปพลิเคชันของ AppReleaseJ ในปัจจุบัน

จากขั้นตอนตาม flowchart ข้างต้น เป็นการดำเนินการ release แอปพลิเคชันไป everyone เอกสารซึ่งก่อนจะถูก release ไป everyone จำเป็นต้องถูก release ไป internal ก่อนตามลำดับขั้นตอนการดำเนินการ ไม่ว่าจะเป็นกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

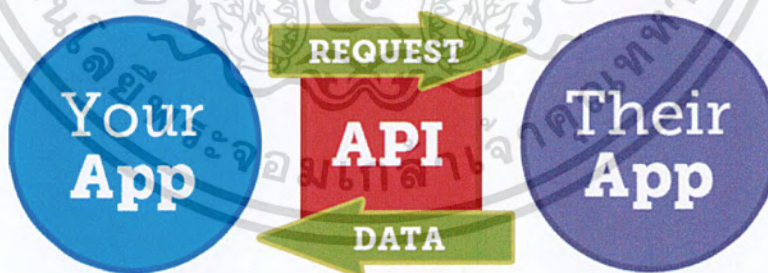
## 2.2 ปัญหาที่พบในปัจจุบัน

- ในการ Release เพื่อให้ได้รับการอนุมัติในแต่ละครั้งนั้น จะออกเป็น version ใหม่ขึ้นมา ถ้าผู้พัฒนาไม่ได้ทำการเขียนอธิบายว่าได้ทำการพัฒนาในส่วนใดไปบ้าง (Release Note) ผู้ตรวจสอบและผู้อนุมัติจะต้องเสียเวลาไปไต่ดู
- ถ้ามีการกำหนดวันให้ปล่อยแอปพลิเคชันที่ต้องการให้ผู้ใช้ทั่วไปใช้งาน แล้วผู้มีสิทธิ์อนุมัติลืมทำการอนุมัติ จะทำให้ผู้ใช้ทั่วไปได้รับแอปพลิเคชันใหม่ล่าช้าได้ เนื่องจากเป็นระบบที่ต้องทำการอนุมัติแอปพลิเคชันแบบ manual ทำให้ software product ขาดความน่าเชื่อถือได้
- ในการแสดงหน้าต่างของสถานะการอนุมัติปัจจุบันทำให้นักพัฒนายากต่อการค้นหา เนื่องจากไม่มีการแสดงสถานะการอนุมัติแบบเป็นลำดับขั้น

## 2.3 เทคโนโลยีที่ใช้ในการพัฒนาระบบ

### 2.3.1 Application Programming Interface (API)

API คือระบบบริการข้อมูลกลางระหว่าง Client และการทำงานฝั่ง Server โดยหน้าที่หลักคือคอยรับคำสั่งจากฝั่ง Client จากแอปพลิเคชันต่างๆ เมื่อเกิดคำร้องขอจาก Client แล้ว API จะรับคำสั่งนั้นไปประมวลผลเรียกว่า Request และส่งข้อมูลที่ตรงกับคำร้องขอกลับไปให้ส่วน Client เรียกว่า Response โดยข้อมูลที่ถูกส่งกลับมาในรูปแบบของ JSON, XML เป็นต้น

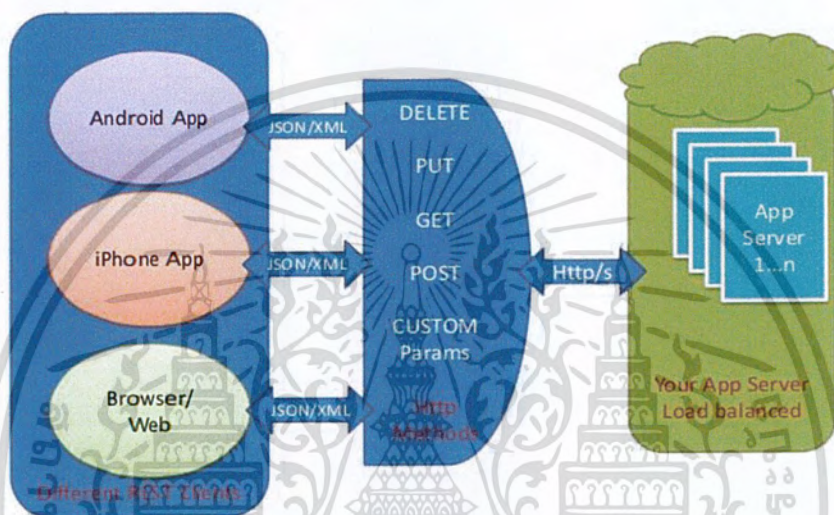


รูปที่ 2.3 Application Programming Interface (API)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.2 REST API (Representational State Transfer)

REST API คือ วิธีในการสร้าง Web Service รูปแบบหนึ่งที่ใช้ HTTP Method (GET, POST, PUT, DELETE) ในการทำงาน และส่งผลกลับมาในรูปแบบของ JSON หรือ XML ส่งผลให้สามารถรับส่งข้อมูลข้ามแพลตฟอร์มได้สะดวก โดยการเรียกใช้ผ่าน HTTP Protocol ที่ใช้ในการเรียกเว็บไซต์



รูปที่ 2.4 REST API Architecture

คำสั่งที่ใช้งานตาม HTTP Verbs ดังนี้

- GET ดึงข้อมูลภายใน URL ที่กำหนด
- POST สร้างข้อมูล
- PUT แก้ไขข้อมูล
- DELETE ลบข้อมูล

โดยข้อดีของการมี Web Service สำหรับใช้งาน API

- 1) เพื่อเพิ่มความสะดวกในการพัฒนาที่แยกการพัฒนาออกเป็น 2 ส่วนคือ Front-End และ Back-End การเพิ่ม/ลดในงานทำได้สะดวกและยืดหยุ่นมากขึ้น
- 2) ช่วยให้นักพัฒนา Front-End และ Back-End ทำงานได้สะดวกมากขึ้น
- 3) Web Service เดียวสามารถใช้งานได้ทั้งในเว็บแอปพลิเคชัน และโมบายแอปพลิเคชัน เพราะข้อมูลที่ส่งกลับมาเป็น JSON ทำให้สามารถไปพัฒนาต่อได้ง่าย
- 4) ทำให้เกิดการกระจายข้อมูลไม่เพียงแต่แอปพลิเคชันเดียวเท่านั้นที่สามารถใช้บริการได้

เอกสารนี้เป็นเอกสารที่เผยแพร่โดยคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.3 ภาษา HTML

HTML ย่อมาจาก Hyper Text Markup Language คือภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของเอกสารบน website หรือที่เรียกว่าเว็บเพจ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้าน Software ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้ หรือที่เรียกว่า HTML Application

HTML เป็นภาษาประเภท Markup สำหรับการการสร้างเว็บเพจ โดยใช้ภาษา HTML สามารถทำโดยใช้โปรแกรม Text Editor ต่างๆ เช่น Notepad, Editplus หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Microsoft FrontPage, Dream Weaver ซึ่งอำนวยความสะดวกในการสร้างหน้า HTML ส่วนการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้โปรแกรม web browser เช่น IE Microsoft Internet Explorer (IE), Mozilla Firefox, Safari, Opera, และ Netscape Navigator เป็นต้น

### 2.3.4 MongoDB

MongoDB เป็น open-source document database โดยเป็นฐานข้อมูลแบบ NoSQL คือไม่มี relation (ความสัมพันธ์) ของตารางแบบ SQL ทั่วไป แต่จะเก็บข้อมูลเป็นแบบ JSON (JavaScript Object Notation) แทนการบันทึกข้อมูลทุกๆ record ใน MongoDB เรียกว่า Document ซึ่งจะเก็บค่าเป็น key และ value ตัวอย่างดังรูปที่ 2.5

```
{
  "_id": ObjectId("554b8ee746e04bc5503aef47"),
  "name": "Chai"
}
```

รูปที่ 2.5 การเก็บค่าเป็น key และ value ของ JSON

และการเก็บข้อมูล document ใน MongoDB จะถูกเก็บไว้ใน Collections (เปรียบเทียบกับ Table ใน Relational Database ทั่วไป) แต่แตกต่างกันที่ collection ไม่จำเป็นต้องมี schema เหมือนกันก็สามารถบันทึกข้อมูลได้

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น เมื่อผู้จัดทำเอกสารได้เห็นว่าเอกสารนี้มีความสำคัญและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.5 Postman

Postman คือเครื่องมือสำหรับช่วยในการพัฒนา API (API Developers) ทดสอบการทำงานของ Service รวมถึงการ Mock Service ซึ่งสามารถช่วยลดระยะเวลาและขั้นตอนลงได้ โดย Postman เป็นที่นิยมอย่างมาก เนื่องจากมี User Interface ที่สวยและใช้งานง่ายกว่า Tools อื่น ๆ โดยผู้ใช้งานไม่จำเป็นต้องมีความรู้เรื่องภาษาโปรแกรมมิ่งก็สามารถใช้งานได้

ความสามารถหลักของ Postman ประกอบไปด้วย

- การทดสอบ API
- การ Test API แบบ Automated
- การ Mock Service
- การเขียน Test Script
- สามารถใช้งานได้บน Chrome โดยผู้ใช้ไม่จำเป็นต้องติดตั้งโปรแกรมลงบนเครื่อง

### 2.3.6 Sysinternals Process Explorer

Process Explorer เป็นเครื่องมือจาก Sysinternals ที่เป็นหน่วยงานหนึ่งของ Microsoft ใช้สำหรับตรวจสอบดูการทำงานของระบบปฏิบัติการ Windows เป็นเครื่องมือที่ใช้แสดง Process ที่ทำงานอยู่ ทั้งด้านเบื้องหน้า (Foreground) และเบื้องหลัง (Background) นอกจากนี้ยังสามารถแสดงรายละเอียดต่างๆ เกี่ยวกับ Handles และ DLLs ที่ Process ทำการเปิดใช้งานอยู่ ซึ่ง Process Explorer ในเวอร์ชันปัจจุบัน ได้แก่เวอร์ชัน 12.04 มีการปรับปรุงด้าน GUI , Performance ให้ดีขึ้น ตัวโปรแกรมใช้งานง่าย สามารถใช้ได้โดยไม่ต้องเป็น Administrator หรือผู้ดูแลระบบ รองรับการทำงานบน Windows XP ขึ้นไปทั้ง 32 บิต และ 64 บิต สำหรับเครื่องที่เป็น Client และ เครื่องที่เป็น Server ตั้งแต่ Windows Server 2003 ขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



### 2.3.9 Jenkins

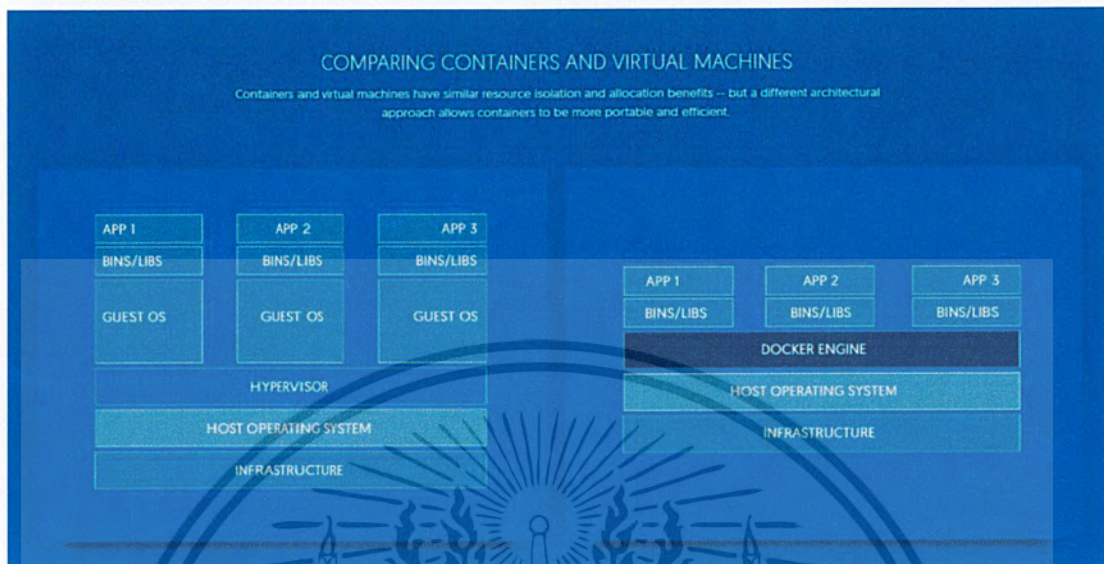
Jenkins คือ open source continuous integration tools เขียนโดย java สนับสนุน source code management เช่น git subversion cvs เป็นต้น เป็น tools ที่สะดวก เมื่อทำการ push code ผ่านทาง git (ถ้าทำการเชื่อมต่อ git hook ไปยัง Jenkins server) ตัว Jenkins ก็จะมีการ build โปรเจคให้ด้วยอัตโนมัติตามที่ได้ทำการเขียนไว้



### 2.3.10 Docker

Docker คือ engine ตัวหนึ่งที่มีการทำงานในลักษณะจำลองสภาพแวดล้อมขึ้นมาบนเครื่อง server เพื่อใช้ในการ run service ที่ต้องการ มีการทำงานคล้ายคลึงกับ Virtual Machine เช่น VMWare, VirtualBox, XEN, KVM แต่ข้อแตกต่างที่ชัดเจนคือ Virtual Machine ที่รู้จักกันก่อนหน้านี้เป็นการจำลองทั้ง OS เพื่อใช้งานและหากต้องการใช้งาน service ใดๆ จึงทำการติดตั้งเพิ่มเติมบน OS นั้นๆ แต่สำหรับ docker แล้วจะใช้ container ในการจำลองสภาพแวดล้อมขึ้นมา เพื่อใช้งานสำหรับ 1 service ที่ต้องการใช้งานเท่านั้น โดยไม่ต้องมีส่วนของ OS เข้าไปเกี่ยวข้องเหมือน Virtual Machines อื่นๆ ตัวอย่างดังรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## รูปที่ 2.8 การเปรียบเทียบ container กับ virtual machines

Docker นั้นเป็นที่รู้จักกันอย่างแพร่หลายในช่วง 1-2 ปีที่ผ่านมา เนื่องจากสามารถใช้งานได้สะดวกและตอบสนองความต้องการของ ผู้พัฒนาโปรแกรม (Developer) หรือ ผู้ดูแลระบบ (System admin)

### Docker image

Docker image เป็นเหมือนตัวต้นแบบของ container ซึ่งภายในจะประกอบด้วย application ต่างๆ ที่มีการติดตั้งไว้เพื่อใช้งานสำหรับ service นั้นๆ รวมทั้งมีการ config ค่าต่างๆ ไว้เรียบร้อยแล้ว จากนั้นก็นำมาสร้างเป็น docker image บน registry เพื่อนำไปใช้งาน ทั้งนี้ผู้ใช้งานสามารถยังสร้าง docker image สำหรับใช้งานเองได้อีกด้วย

### Docker container

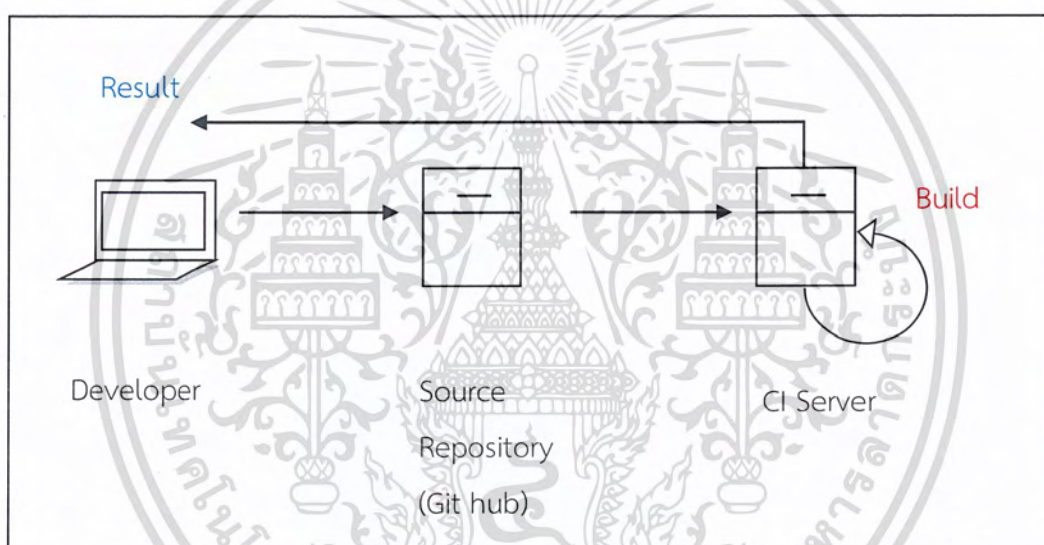
Docker container สามารถมองได้เสมือนกล่อง ซึ่งนำ docker image มาติดตั้ง เพื่อให้สามารถใช้งาน service ที่ต้องการจาก image นั้นๆ ได้ โดยใน container แต่ละตัวจะมีการใช้งาน RAM, CPU, ไฟล์ config ต่างๆ เป็นของแต่ละ container เอง และยังสามารถสั่ง start, stop ได้ที่ container นั้นๆ อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.11 Continuous Integration (CI)

Continuous Integration (CI) คือแนวทางการพัฒนาซอฟต์แวร์ที่อิงกับการรวมโค้ดลงในที่เก็บข้อมูลร่วมกัน โดยการ push code ขึ้น Project แต่ครั้งจะได้รับการยืนยันตนโดยอัตโนมัติ โดยจะรู้ว่าใครเป็นคน push code ชุดนั้น

CI ไม่ได้เป็น Software ที่ใช้สำหรับตรวจสอบ Code ว่า Code แต่ละชุด Build ผ่านหรือไม่ แต่เป็นเหมือนวัฒนธรรมหรือกฎของคนในทีมต่างหาก ที่ต้องตรวจสอบ Code ของ ตัวเองว่ามีจุดบกพร่องตรงไหนที่ทำให้ไม่สามารถ Build Code ผ่านได้ โดยจะเป็นการสร้างวินัยให้คนในทีมและจะช่วยลดเวลาการแก้ Code ที่ Error โดยจะใช้เวลาน้อยลง



รูปที่ 2.9 แนวการพัฒนาแบบ Continuous Integration (CI)

จากรูปที่ 2.9 เมื่อ Developer ทำการพัฒนาหรือเขียนโปรแกรมเรียบร้อยแล้ว จะทำการ push code ดังกล่าวขึ้น Git hub เมื่อถึงเวลาที่กำหนดเครื่อง CI Server จะทำการ Build code อัตโนมัติ และส่งผลลัพธ์ที่ได้จากการ build ไปแสดงให้ Developer ทราบ

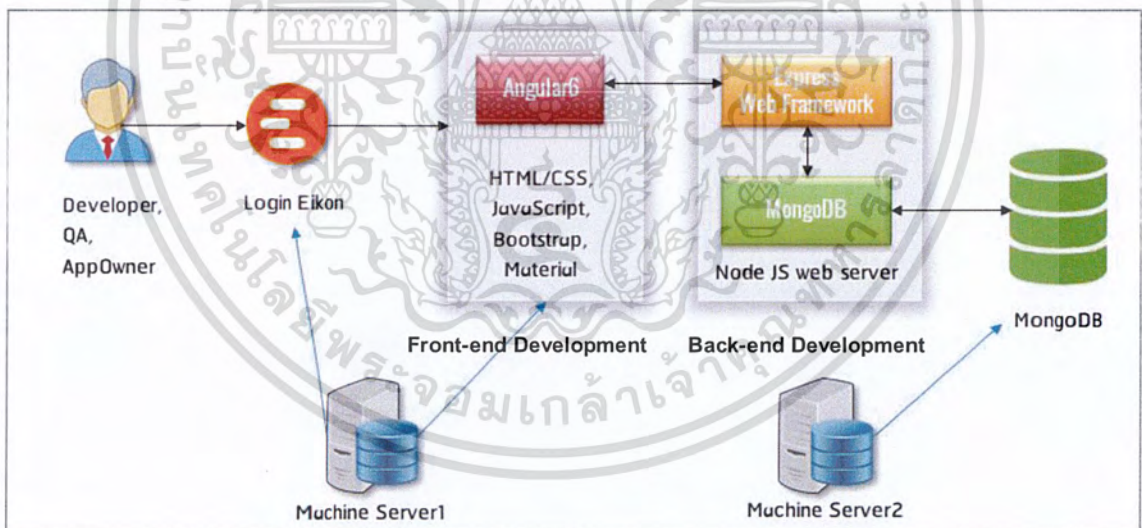
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### ขั้นตอนการพัฒนาระบบ

เนื่องจากปัจจุบันในส่วนของการจัดการแอปพลิเคชันที่อยู่ภายใน software Eikon ทางบริษัท Refinitiv มีระบบการปล่อยแอปพลิเคชันที่สร้างขึ้นโดยใช้ระบบ AppReleaseJ ซึ่งระบบดังกล่าวจะมีขั้นตอนการร้องขอ การอนุมัติ และการยกเลิกโดยการทำเนิการดังกล่าว กระทำโดยใช้วิธีแบบ manual หลายขั้นตอน ซึ่งถ้าผู้มีสิทธิ์อนุมัติทำการอนุมัติ จะทำให้แอปพลิเคชันนั้นๆ ถูกปล่อยออกไปล่าช้าได้ จึงเป็นเหตุให้มีการสร้างระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ขึ้น เพื่อสร้างความสะดวกมากขึ้นในการ release เป็นเสมือนเครื่องมือที่ช่วยให้การอนุมัติแอปพลิเคชันให้เป็นไปโดยอัตโนมัติ

#### 3.1 โครงสร้างโดยรวมของระบบ



รูปที่ 3.1 แผนภาพแสดงโครงสร้างโดยรวมของระบบ

1. เมื่อผู้ใช้เข้าสู่หน้าเว็บไซต์ของระบบ หน้าเว็บไซต์ (Front-end) จะทำการประมวลผลและแสดงผล ทำให้ผู้ใช้สามารถใช้งานระบบได้ตามสิทธิ์ของผู้ใช้
2. มีการขอข้อมูลจาก API เพื่อใช้ในการจัดการระบบคือ การดึงข้อมูลรายละเอียดของแอปพลิเคชัน

ที่มีการ Release เพื่อนำข้อมูลเหล่านี้มาใช้ในแสดงผล การดำเนินการร้องขอให้อนุมัติ และ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า การดำเนินการอนุมัติ มีการใช้ API Web Service ในการจัดการส่วน Service (Back-end) ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. มีการเก็บข้อมูลต่างๆ เช่นข้อมูลแอปพลิเคชัน ข้อมูลผู้ใช้งาน เป็นต้น โดยใช้ MongoDB ซึ่งเป็นฐานข้อมูลแบบ NoSQL

ในส่วนของผู้ใช้งาน 3 ประเภทสามารถใช้งานระบบได้ดังนี้

1) นักพัฒนาแอปพลิเคชัน (Developer)

- สามารถดูสถานะการ Release แอปพลิเคชันได้
- สามารถดูรายชื่อและเวอร์ชันแอปพลิเคชันได้
- สามารถสร้างการ Release ของแอปพลิเคชันได้
- สามารถร้องขอการอนุมัติการเปลี่ยนสิทธิ์ผู้ใช้งานแอปพลิเคชันจากผู้อนุมัติได้
- สามารถทำการเขียนอธิบายว่าได้ทำการพัฒนาในส่วนใดไปบ้าง (Release Note)

2) นักพัฒนา/ผู้บริหารระดับสูง (AppOwner)

- สามารถดูสถานะการ Release แอปพลิเคชันได้
- สามารถดูรายชื่อและเวอร์ชันแอปพลิเคชันได้
- สามารถสร้างการ Release ของแอปพลิเคชันได้
- สามารถดูแอปพลิเคชันที่มีการร้องขอ Release ได้
- สามารถทำการอนุมัติการเปลี่ยนสิทธิ์ผู้ใช้งานแอปพลิเคชันจากผู้ร้องขอได้
- สามารถทำการถอน (withdrawn) Release ได้

3) ผู้ตรวจสอบระบบ (QA)

- สามารถดูสถานะการ Release แอปพลิเคชันได้
- สามารถดูรายชื่อและเวอร์ชันแอปพลิเคชันได้
- สามารถสร้างการ Release ของแอปพลิเคชันได้
- สามารถดูแอปพลิเคชันที่มีการร้องขอ Release ได้
- สามารถร้องขอการอนุมัติการเปลี่ยนสิทธิ์ผู้ใช้งานแอปพลิเคชันจากผู้อนุมัติได้
- สามารถทำการอนุมัติแอปพลิเคชันที่ทำการตรวจสอบได้
- สามารถทำการถอน (withdrawn) Release ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 การวิเคราะห์ระบบ

### 3.2.1 Functional Requirement

เนื่องจากระบบที่พัฒนาขึ้นเป็นระบบที่สร้างขึ้นใหม่จึงจำเป็นต้องออกแบบและวิเคราะห์ระบบทั้งในส่วน Front-end และ Back-end ว่าควรมีฟังก์ชันอะไรบ้างเพื่อให้ตอบสนองความต้องการของบริษัทและสอดคล้องกับความสามารถของระบบ AppReleaseJ

#### การออกแบบ state

ในการออกแบบ state ของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv จะทำการออกแบบโดยอ้างอิงถึง status จริงที่ปรากฏบนระบบ AppreleaseJ ซึ่งระบบ AppreleaseJ มี status ทั้งหมดดังนี้

- Internal Release App Owner Approval  
: สถานะนี้จะแสดงขึ้นหลังจากมีการร้องขอ release
- Internal Release Ops Approval - Primary DC  
: สถานะนี้จะแสดงขึ้นหลังจากผู้อนุมัติทำการกด approve ไป internal
- Internal Release Complete - Primary DC  
: สถานะนี้จะแสดงขึ้นเมื่อระบบทำการ release สู่ data center ที่แรกสำเร็จ
- Internal Release Ops Approval for All DCs  
: สถานะนี้จะแสดงขึ้นหลังจากการ release สู่ data center ที่แรกสำเร็จ เพื่อรระบบ release สู่ data center ทั้งหมดที่เหลืออยู่
- Internal Release Complete - All DCs  
: สถานะนี้จะแสดงขึ้นเมื่อระบบทำการ release สู่ data center สำเร็จครบทุกภูมิภาค
- Everyone Release App Owner Approval  
: สถานะนี้จะแสดงขึ้นหลังจาก release ถูกปล่อยสู่ Internal
- Everyone Release Ops Approval - Primary DC  
: สถานะนี้จะแสดงขึ้นหลังจากผู้อนุมัติทำการกด approve ไป everyone
- Everyone Release Complete - Primary DC  
: สถานะนี้จะแสดงขึ้นเมื่อระบบทำการ release สู่ data center ที่แรกสำเร็จ
- Everyone Release Ops Approval for All DCs  
: สถานะนี้จะแสดงขึ้นหลังจากการ release สู่ data center ที่แรกสำเร็จ เพื่อรระบบ release สู่ data center ทั้งหมดที่เหลืออยู่
- Complete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากกระบวนการการทำงานของระบบ AppreleaseJ ทำให้มีการออกแบบระบบโดยให้หน้าเว็บแสดง state ทั้งหมด 6 state ดังนี้

State 1: Wait for QA Test ( รอ QA กด sign off)

State 2: Wait for AppOwner approve ( รอผู้มีสิทธิ์อนุมัติกด approve )

State 3: Wait for TR Internal ( รอระบบดำเนินการ release ไป TR Internal )

State 4: Internal ( แสดงสถานะ Internal )

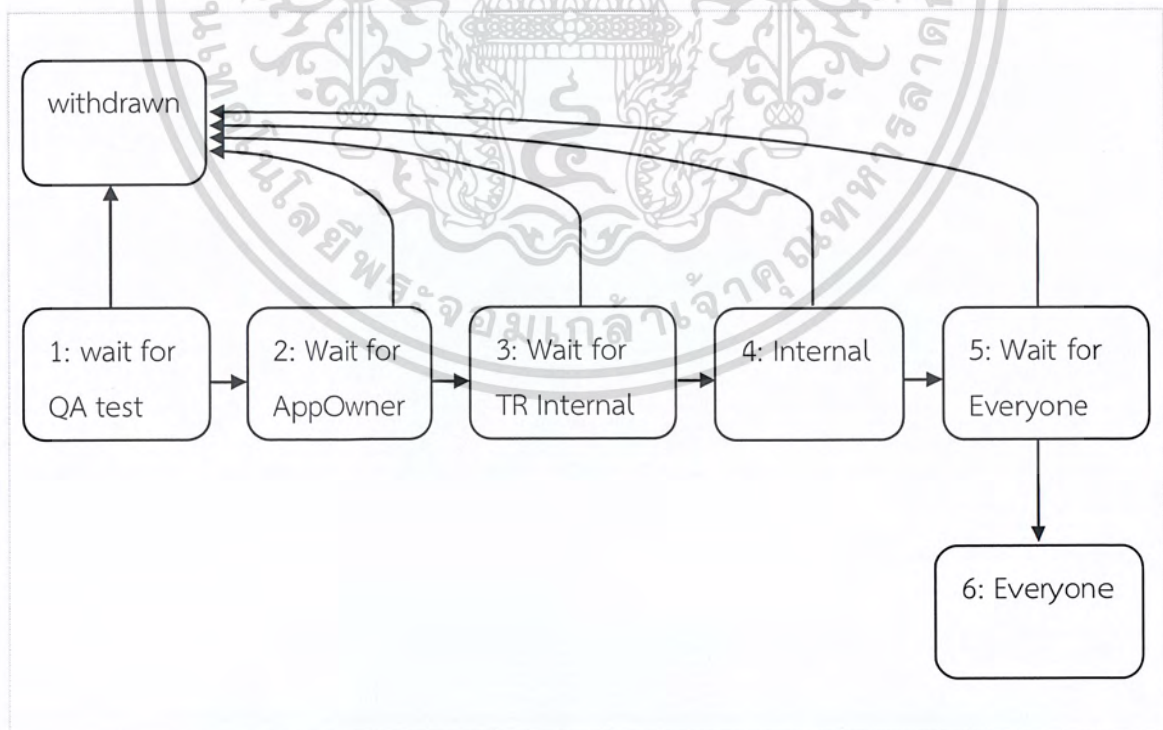
State 5: Wait for Everyone ( รอระบบดำเนินการ release ไป Everyone )

State 6: Everyone ( แสดงสถานะ Everyone )

ซึ่งในช่วงเวลาการ release แอปพลิเคชันนั้น AppOwner หรือ QA สามารถทำการถอนการ release ได้ โดยกดที่ปุ่ม withdrawn

การออกแบบกระบวนการการเปลี่ยน state

ในการเปลี่ยน state ของแอปพลิเคชันในระบบที่สร้างนั้น จำเป็นต้องมีความสัมพันธ์กับ status ของระบบ AppReleaseJ สามารถออกแบบ workflow state ได้ดังนี้



รูปที่ 3.2 workflows การเปลี่ยน state ของระบบเว็บแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 รายละเอียดการเปลี่ยน state ของแอปพลิเคชันในระบบ

Before State	Action	After State	What happened
-	Create release application	1	User can click button for next state
1	QA or App Owner click sign off button	2	System will automatically create Release ticket in another system (AppReleaseJ)
2	App Owner click approve button	3	Auto change state by monitoring the state from another system (AppReleaseJ)
3	Check status from AppReleaseJ, this state will be change when the current status is <b>Internal Release Complete - All DCs</b>	4	-
4	Check status from AppReleaseJ, if current status is <b>Everyone Release App Owner Approval</b> System will automatically app owner approve for everyone request	5	Auto change state by monitoring the state from another system (AppReleaseJ)
5	Check status from AppReleaseJ, this state will be change when the current status is <b>Everyone Release Complete - All DCs</b>	6	-
1,2,3,4,5	QA ,App Owner click button withdrawn	-	All state in the AppreleaseJ set to withdraw

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 ข้อมูลจากระบบอื่นที่นำมาใช้ในระบบ

เนื่องจากเป็นระบบใหม่ในส่วนของ service จึงจำเป็นต้องมีการออกแบบให้สอดคล้องกับระบบ AppreleaseJ ซึ่งในระบบใหม่นี้มีการออกแบบ service ได้ดังนี้

- service สำหรับแสดงสถานะการ release ทั้งหมด
- service สำหรับแสดงรายละเอียดของแอปพลิเคชันแต่ละตัวที่ release
- service สำหรับการสร้าง release
- service สำหรับการเปลี่ยนสถานะ
- service สำหรับการ login/logout

ในส่วนของ service สำหรับการเปลี่ยนสถานะ สามารถแบ่งออกเป็น 2 ส่วน คือ ส่วนที่เปลี่ยนสถานะตามผู้ใช้ระบบ กับ ส่วนที่เปลี่ยนสถานะอัตโนมัติ ซึ่งส่วนที่เปลี่ยนสถานะตามผู้ใช้ระบบสามารถเปลี่ยนสถานะได้ดังนี้

- เปลี่ยนสถานะจาก wait for QA test ไปเป็น wait for AppOwner Approve
- เปลี่ยนสถานะจาก wait for AppOwner Approve ไปเป็น wait for TR Internal
- เปลี่ยนสถานะไปเป็น withdrawn

ในส่วนที่เปลี่ยนสถานะอัตโนมัติสามารถเปลี่ยนสถานะได้ดังนี้

- เปลี่ยนสถานะจาก wait for TR Internal ไปเป็น Internal
- เปลี่ยนสถานะจาก Internal ไปเป็น wait for Everyone
- เปลี่ยนสถานะจาก wait for Everyone ไปเป็น Everyone

ซึ่งในการออกแบบระบบจำเป็นต้องนำข้อมูลจากระบบ AppReleaseJ มาใช้ร่วมด้วย

ข้อมูลจากระบบอื่น (API ของระบบ AppReleaseJ) ที่นำมาใช้ในระบบระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv คือ ข้อมูลรายละเอียด API ของการร้องขออนุมัติ (AddNewRule) ข้อมูลรายละเอียด API ของการอนุมัติ ข้อมูลรายละเอียด API ของการ withdraw และข้อมูลรายละเอียด API ของการแสดงสถานะแอปพลิเคชันใน AppreleaseJ

### 3.2.3 Use Case Diagram



รูปที่ 3.3 Use Case Diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 3.3 สามารถอธิบายได้ดังนี้

1. Login คือ การลงชื่อเข้าใช้งานระบบเว็บไซต์เพื่อสามารถมีสิทธิ์เข้าถึงการใช้งานเพิ่มขึ้น
2. View State Release คือ การดูสถานะของแอปพลิเคชันที่มีการ release ทั้งหมด โดยจะมีการแสดงสถานะทั้งหมด 7 สถานะดังนี้
  - Wait for QA test คือ สถานะรอ QA ตรวจสอบระบบและ Sign off ระบบ
  - Wait for AppOwner คือ สถานะรอ AppOwner อนุมัติ
  - Wait for TR Internal คือ สถานะรอแอปพลิเคชัน release ไป TR Internal
  - Internal คือ สถานะการ release ไป internal สำเร็จ
  - Wait for Everyone คือ สถานะรอแอปพลิเคชัน release ไป Everyone
  - Everyone คือ สถานะการ release ไป everyone สำเร็จ
  - Withdrawn คือ สถานะการถอน/ยกเลิกการ release
3. View Detail Release คือ การดูรายละเอียดของแอปพลิเคชันที่มีการ release
4. Create Release คือ การสร้างรายการแอปพลิเคชันที่ต้องการร้องขอ release
5. Wait for QA Sign off คือ เมื่อมีการร้องขอ release สถานะเริ่มต้นคือ Wait for QA Test เป็นการรอให้ QA ตรวจสอบและ Sign off ระบบ
6. Signed off Release คือ การกด Signed off release แอปพลิเคชันที่มีการร้องขอ
7. Wait for AppOwner คือ เมื่อมีการกด Signed off release สถานะจะเปลี่ยนเป็น Wait for AppOwner เป็นการรอให้ AppOwner อนุมัติแอปพลิเคชันที่มีการร้องขอ
8. Create Ticket คือ การสร้างเลข ticket สำหรับแอปพลิเคชันที่ร้องขอให้ Approve
9. Approved Release คือ การกด Approved release แอปพลิเคชันที่มีการร้องขอ
10. Withdrawn Release คือ การกด Withdrawn release แอปพลิเคชันที่ต้องการยกเลิก
11. Search AppNames คือ การค้นหารายชื่อแอปพลิเคชันที่ต้องการ
12. Sort State in Ascending or Descending order คือ การเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชันจากมากไปน้อยหรือน้อยไปมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 3.3 Use case diagram ของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv สามารถแบ่งฟังก์ชันที่มีทั้งหมดได้ดังนี้

1) Create release : การสร้างรายชื่อแอปพลิเคชันที่ต้องการขอ release บนหน้าระบบ

ตารางที่ 3.2 Use Case Documentation ของฟังก์ชัน Create release

หัวข้อ	คำอธิบาย	
Use Case Name :	Create release	
Use Case ID :	01	
Action :	Developer, QA, AppOwner	
Description :	เป็นระบบการสร้างรายชื่อแอปพลิเคชันที่ต้องการขออนุมัติ	
Pre-Conditions :	มีแบบฟอร์มต่างๆให้กรอกรายละเอียดตามแบบฟอร์ม	
Post-Conditions :	มีการแสดงผลลัพธ์ว่าผู้ใช้สร้างรายชื่อได้สำเร็จหรือไม่	
Flow :	User	System
	กรอกข้อมูลตามแบบฟอร์ม ที่ผู้ใช้ต้องการสร้าง	ทำการเก็บข้อมูล รายละเอียดการสร้าง และ แสดงผลในหลักของระบบ

2) View state release : การแสดงผลในส่วนการหน้าหลัก มีการแสดงรายชื่อ เวอร์ชัน และสถานะของแอปพลิเคชัน

ตารางที่ 3.3 Use Case Documentation ของฟังก์ชัน View state release

หัวข้อ	คำอธิบาย	
Use Case Name :	View state release	
Use Case ID :	02	
Action :	Developer, QA, AppOwner	
Description :	เป็นระบบที่แสดงรายชื่อแอปพลิเคชันที่มีการ release	
Pre-Conditions :	สามารถเลือกดูรายชื่อแอปพลิเคชันที่ต้องการได้	
Post-Conditions :	มีการแสดงรายชื่อแอปพลิเคชันทั้งหมดที่มีการ release	
Flow :	User	System
	แสดงข้อมูลแอปพลิเคชัน ต่างๆที่มีการร้องขอ release	เรียกข้อมูลแอปพลิเคชัน จากระบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่จำกัดเท่านั้น ไม่อนุญาตให้มีการเผยแพร่ข้อมูลหรือข้อมูลใดๆ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของสิทธิ์ที่มีกรรมสิทธิ์

- 3) View detail release : ดูรายละเอียดของแอปพลิเคชันที่มีการ release แสดงชื่อแอปพลิเคชัน เวอร์ชันแอปพลิเคชัน release note และแสดงวันที่เริ่มแรกที่ทำกร create release เป็นต้น

#### ตารางที่ 3.4 Use Case Documentation ของฟังก์ชัน View detail release

หัวข้อ	คำอธิบาย	
Use Case Name :	View detail release	
Use Case ID :	03	
Action :	Developer, QA, AppOwner	
Description :	เป็นระบบแสดงรายละเอียดแอปพลิเคชันที่มีการ release	
Pre-Conditions :	สามารถเลือกดูรายละเอียดของแอปพลิเคชันที่ต้องการได้	
Post-Conditions :	มีการแสดงรายละเอียดของแอปพลิเคชันที่เลือก	
Flow :	User	System
	ทำการคลิกรายชื่อของแอปพลิเคชันที่ต้องการดูรายละเอียด	ทำการเรียกข้อมูลแอปพลิเคชันจากระบบฐานข้อมูลมาแสดงบนหน้าระบบ

- 4) Signed off release : การยินยอมให้การร้องขอดำเนินต่อไป โดยผู้มีสิทธิ์กดปุ่ม sign off จะต้องตรวจสอบ release

#### ตารางที่ 3.5 Use Case Documentation ของฟังก์ชัน Signed off release

หัวข้อ	คำอธิบาย	
Use Case Name :	Signed off release	
Use Case ID :	04	
Action :	QA, AppOwner	
Description :	การยินยอมให้การร้องขอ release ดำเนินต่อไป	
Pre-Conditions :	เลือกแอปพลิเคชันที่ต้องการกด sign off	
Post-Conditions :	แสดงผลลัพธ์ว่าสำเร็จ และสถานะเปลี่ยนเป็น state ถัดไป	
Flow :	User	System
	ทำการเลือกคลิกปุ่ม signed off ในตำแหน่งของแอปพลิเคชันที่ต้องการ	ส่งข้อมูลการขอลปล่อยแอปพลิเคชันดังกล่าวไปยังระบบอื่น และทำการอัปเดตสถานะใหม่บนฐานข้อมูล

- 5) Approved release : การอนุมัติแอปพลิเคชันที่มีการร้องขอ release

ตารางที่ 3.6 Use Case Documentation ของฟังก์ชัน Approved release

หัวข้อ	คำอธิบาย	
Use Case Name :	Approved release	
Use Case ID :	05	
Action :	AppOwner	
Description :	การอนุมัติการร้องขอ release ให้สามารถปล่อยแอปพลิเคชันดังกล่าวสู่ everyone ได้	
Pre-Conditions :	เลือกแอปพลิเคชันที่ต้องการกด Approve	
Post-Conditions :	แสดงผลลัพธ์ว่าสำเร็จ และสถานะเปลี่ยนเป็น state ถัดไป	
Flow :	User	System
	ทำการเลือกคลิกปุ่ม Approved ในตำแหน่งของแอปพลิเคชันที่ต้องการ	ส่งข้อมูลการขออัปเดตแอปพลิเคชันดังกล่าวไปยังระบบอื่น และทำการอัปเดตสถานะใหม่บนฐานข้อมูล

- 6) Withdrawn release : การถอนแอปพลิเคชันที่มีการร้องขอ release

ตารางที่ 3.7 Use Case Documentation ของฟังก์ชัน Withdrawn release

หัวข้อ	คำอธิบาย	
Use Case Name :	Withdrawn release	
Use Case ID :	06	
Action :	QA, AppOwner	
Description :	การถอนการร้องขอ release หรือ การยกเลิกการ release	
Pre-Conditions :	เลือกแอปพลิเคชันที่ต้องการกด Withdraw	
Post-Conditions :	แสดงผลลัพธ์ว่าสำเร็จ และสถานะเปลี่ยนเป็น withdrawn	
Flow :	User	System
	ทำการเลือกคลิกปุ่ม Withdrawn ในตำแหน่งของแอปพลิเคชันที่ต้องการ	ส่งข้อมูลการ withdraw ไปยังระบบอื่น และอัปเดตสถานะใหม่บนฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับภายในระบบเพื่อการใช้งานเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาหรือข้อมูลใดๆ ของเอกสารฉบับนี้โดยไม่ได้รับอนุญาต

## 7) AppName Search : การค้นหารายชื่อแอปพลิเคชันที่ต้องการ

ตารางที่ 3.8 Use Case Documentation ของฟังก์ชัน AppName Search

หัวข้อ	คำอธิบาย	
Use Case Name :	AppName search	
Use Case ID :	07	
Action :	Developer, QA, AppOwner	
Description :	การค้นหารายชื่อแอปพลิเคชันที่มีการร้องขอ release	
Pre-Conditions :	พิมพ์ชื่อแอปพลิเคชันที่ต้องการค้นหา	
Post-Conditions :	แสดงผลลัพธ์ชื่อ เวอร์ชัน และสถานะของแอปพลิเคชัน ที่ทำการค้นหาบนหน้าหลัก	
Flow :	User	System
	ทำการพิมพ์ชื่อแอปพลิเคชันที่ต้องการค้นหา	Filter รายการแอปพลิเคชันตามชื่อที่ผู้ใช้ทำการค้นหา

## 8) Sort by state : การเรียงลำดับการแสดงผลหน้าหลักตามจำนวนสถานะที่ดำเนินการอยู่ในปัจจุบันของแอปพลิเคชัน โดยเรียงตามจำนวนสถานะจากมากไปน้อยหรือจากน้อยไปมาก

ตารางที่ 3.9 Use Case Documentation ของฟังก์ชัน Sort by state

หัวข้อ	คำอธิบาย	
Use Case Name :	Sort by state	
Use Case ID :	08	
Action :	Developer, QA, AppOwner	
Description :	การอนุมัติการร้องขอ release ให้สามารถปล่อยแอปพลิเคชันดังกล่าวสู่ everyone ได้	
Pre-Conditions :	เลือกแอปพลิเคชันที่ต้องการกด Approve	
Post-Conditions :	แสดงผลลัพธ์ว่าสำเร็จ และสถานะเปลี่ยนเป็น state ถัดไป	
Flow :	User	System
	ทำการเลือกคลิกปุ่ม Sort by state	เรียงลำดับการแสดงรายชื่อแอปพลิเคชันตามจำนวนสถานะที่ดำเนินการอยู่ในปัจจุบันของแอปพลิเคชัน

## 9) Login : การลงชื่อเข้าใช้งานระบบ

ตารางที่ 3.10 Use Case Documentation ของฟังก์ชัน Login

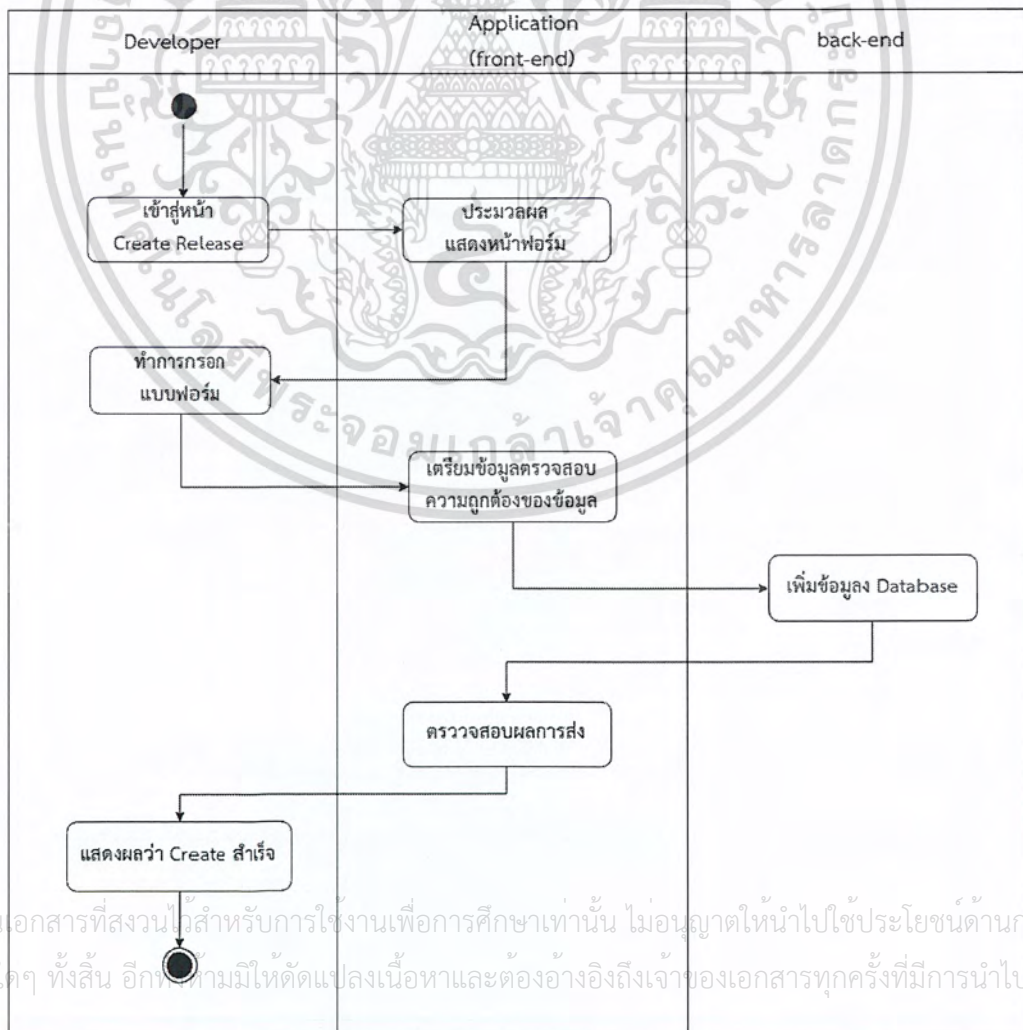
หัวข้อ	คำอธิบาย	
Use Case Name :	Login	
Use Case ID :	09	
Action :	Developer, QA, AppOwner	
Description :	การลงชื่อเข้าใช้งานระบบโดยใช้ username/password ของ Eikon เพื่อทำการร้องขอหรืออนุมัติแอปพลิเคชัน	
Pre-Conditions :	เลือกปุ่ม login บนหน้าเว็บไซต์	
Post-Conditions :	เมื่อลงชื่อเข้าใช้งานสำเร็จ ระบบจะแสดงหน้าหลักโดยแสดงฟังก์ชันต่างๆ ตามสิทธิ์ของผู้ใช้งาน	
Flow :	User	System
	ทำการเลือกคลิกปุ่ม Login จากนั้นทำการกรอก username/password ให้ถูกต้องและทำการกดปุ่ม Sign in	ทำการส่งข้อมูล username และ password ไปยัง API ของ Eikon เพื่อตรวจสอบสิทธิ์ผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 Activity Diagram

ในระบบเว็บแอปพลิเคชันมี Activity Diagram ในการใช้งานเว็บแอปพลิเคชัน ดังนี้

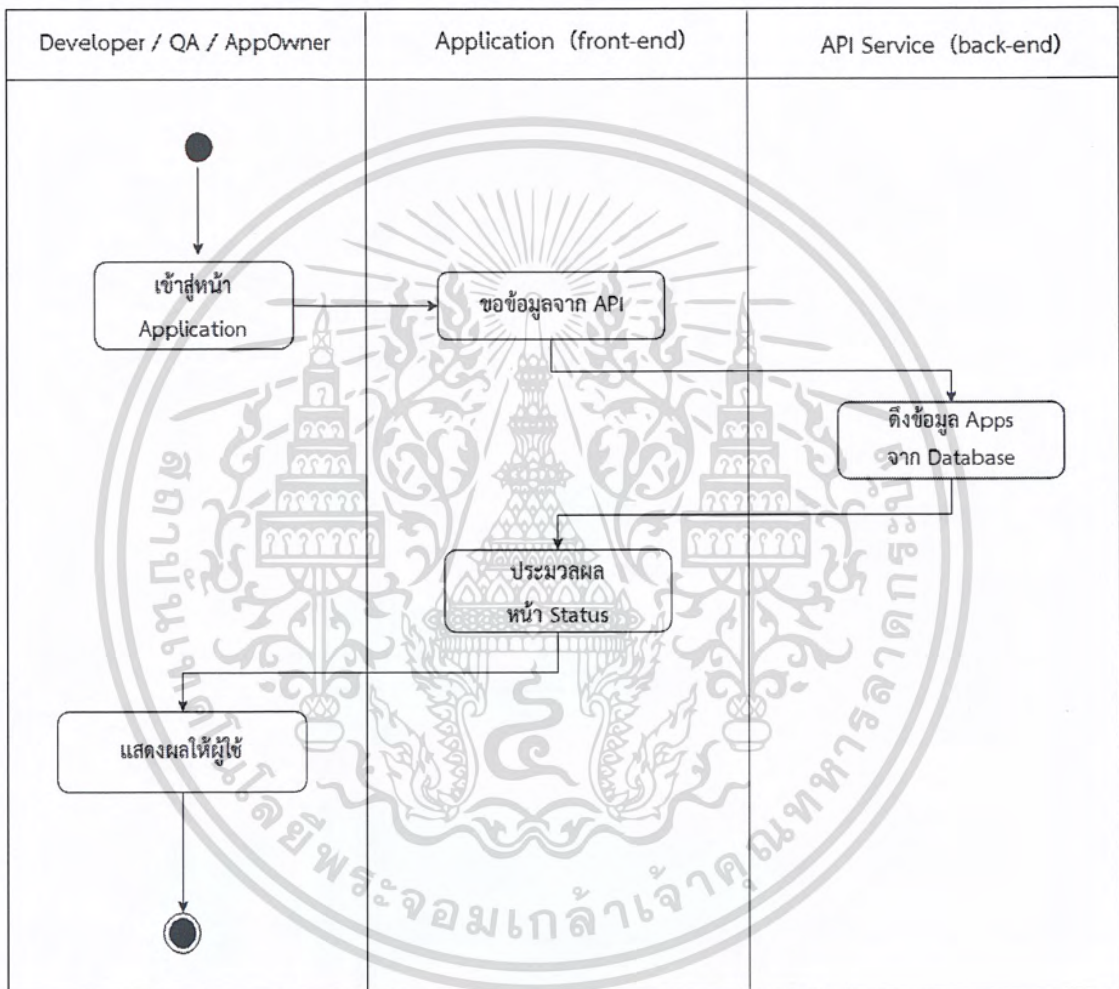
- Activity Diagram ของหน้าเว็บ Create Release ดังรูป 3.4
- Activity Diagram ของหน้าเว็บ View State Release ดังรูป 3.5
- Activity Diagram ของหน้าเว็บ View Detail Release ดังรูป 3.6
- Activity Diagram ของหน้าเว็บ Signed off การร้องขอ Release ดังรูป 3.7
- Activity Diagram ของหน้าเว็บ Approved การร้องขอ Release ดังรูป 3.8
- Activity Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release ดังรูป 3.9
- Activity Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการร้องขอ Release ดังรูป 3.10
- Activity Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน ดังรูป 3.11
- Activity Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ ดังรูป 3.12



รูปที่ 3.4 Activity Diagram ของหน้าเว็บ Create Release

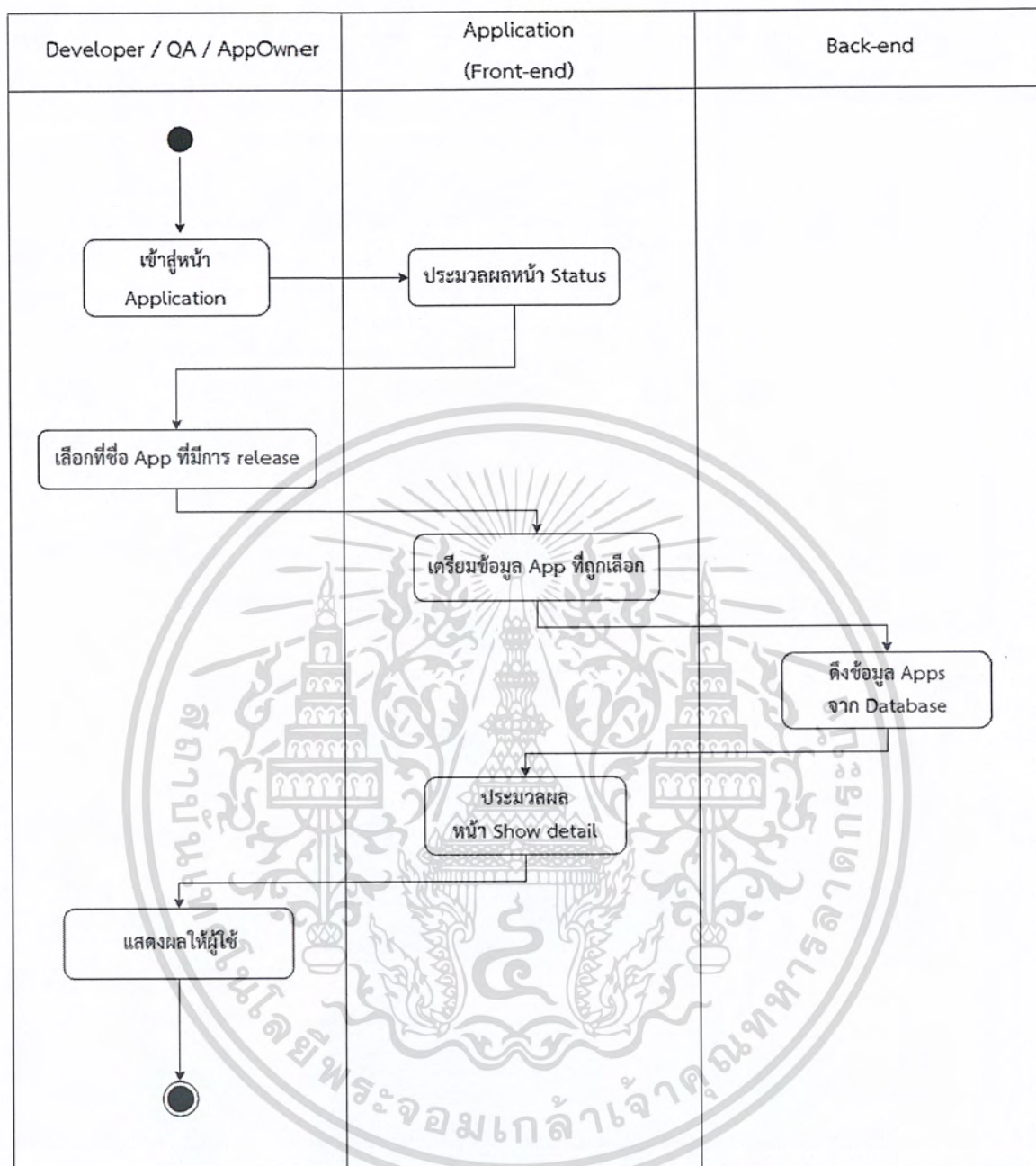
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool มีฟังก์ชัน Create Release ให้ผู้ใช้งานสามารถเลือก โดยเมื่อเลือกที่ปุ่มระบบจะมีหน้าแบบฟอร์มให้กรอก หลังจากเลือกแบบฟอร์มเรียบร้อยแล้ว ผู้ใช้งานกรอกแบบฟอร์มแล้วกดตกลง ระบบทำการส่งข้อมูลรายการไปเก็บในฐานข้อมูล และส่งผลลัพธ์กลับมายังระบบ



รูปที่ 3.5 Activity Diagram ของหน้าเว็บ View State Release

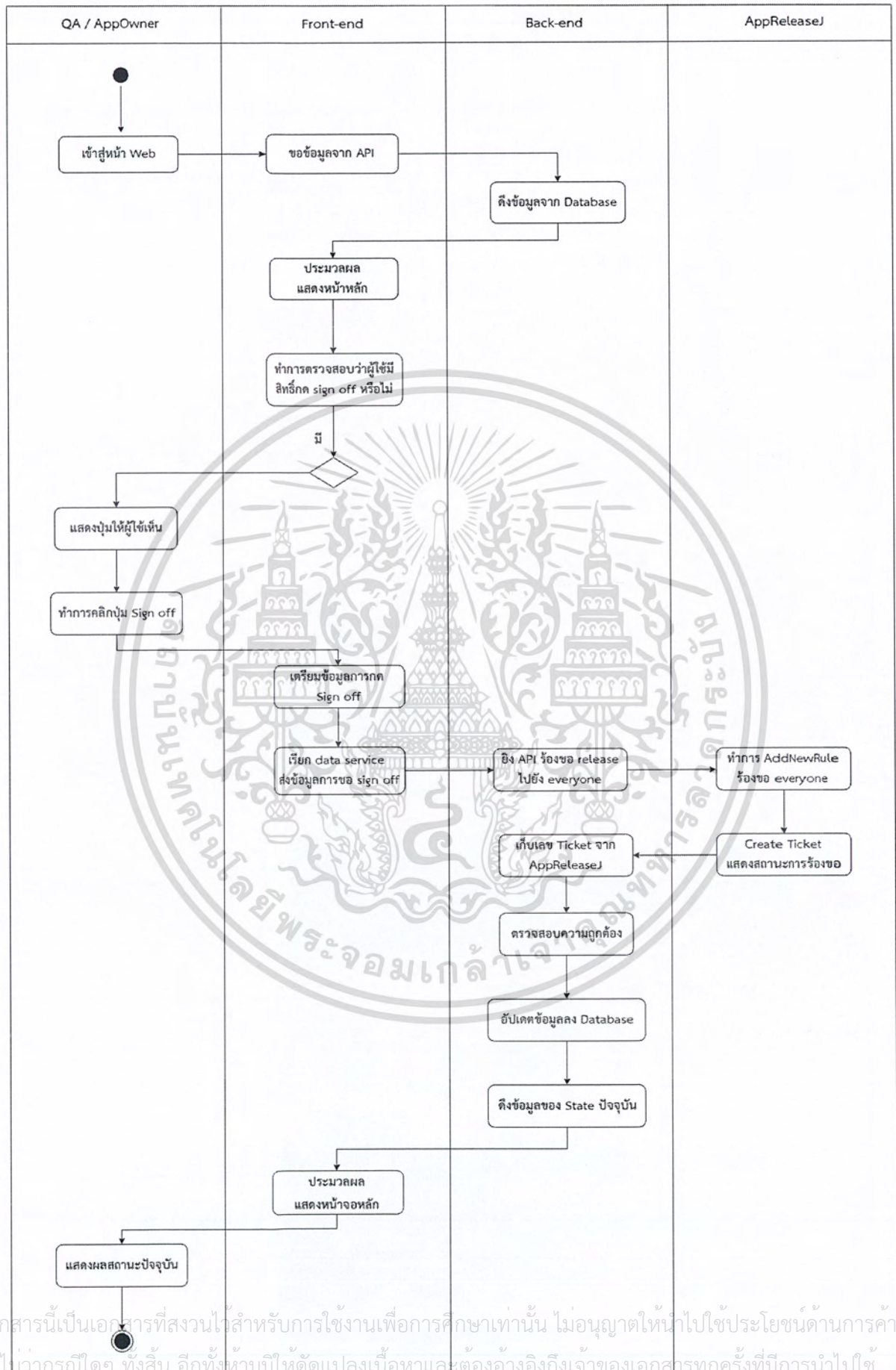
จากรูปที่ 3.5 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชัน View State Release โดยฟังก์ชันดังกล่าวจะทำงานอัตโนมัติเมื่อผู้ใช้งานเข้าสู่หน้าเว็บแอปพลิเคชัน ระบบจะทำการร้องขอข้อมูลของแอปพลิเคชันที่มีการ release จากฝั่ง back-end โดยจะดึงข้อมูลดังกล่าวจากฐานข้อมูล เมื่อได้รับข้อมูลดังกล่าวแล้วระบบจะทำการประมวลผลหน้าเว็บไซต์ และแสดงข้อมูลให้ผู้ใช้งานได้รับทราบ โดยจะมีการแสดงรายชื่อ เวอร์ชัน และสถานะของแต่ละ release ตามลำดับ



รูปที่ 3.6 Activity Diagram ของหน้าเว็บ View Detail Release

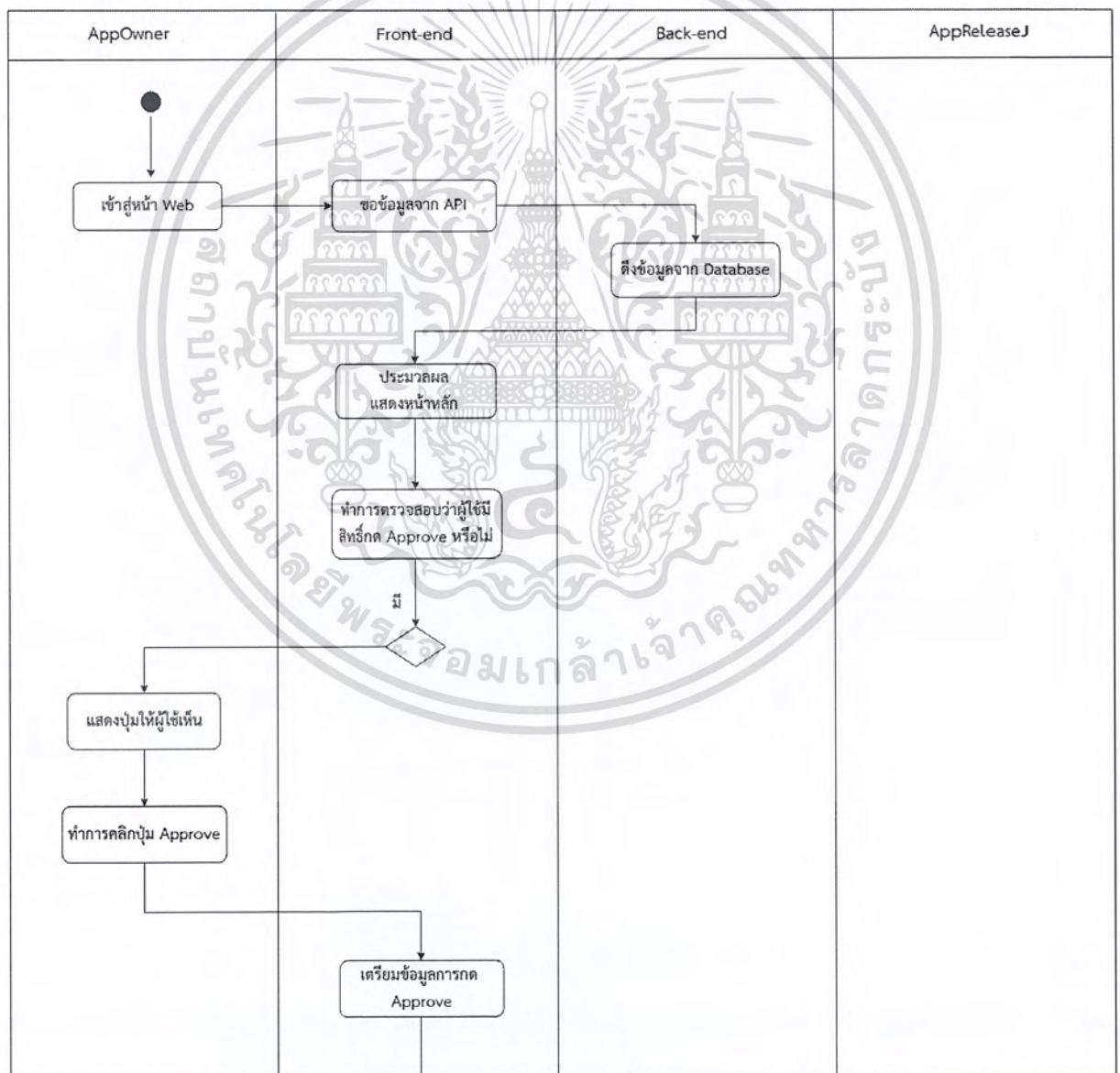
จากรูปที่ 3.6 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชัน View Detail Release โดยเมื่อเลือกกดปุ่มที่ชื่อแอปพลิเคชันที่มีการ release ระบบจะทำการร้องขอข้อมูลของแอปพลิเคชันที่ทำการเลือก จากฝั่ง back-end โดยจะดึงข้อมูลรายละเอียดของแอปพลิเคชันดังกล่าวจากฐานข้อมูล เมื่อได้รับข้อมูลแล้วระบบจะทำการประมวลผลหน้าเว็บไซต์การแสดงผลรายละเอียด และแสดงข้อมูลให้ผู้ใช้งานได้ทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



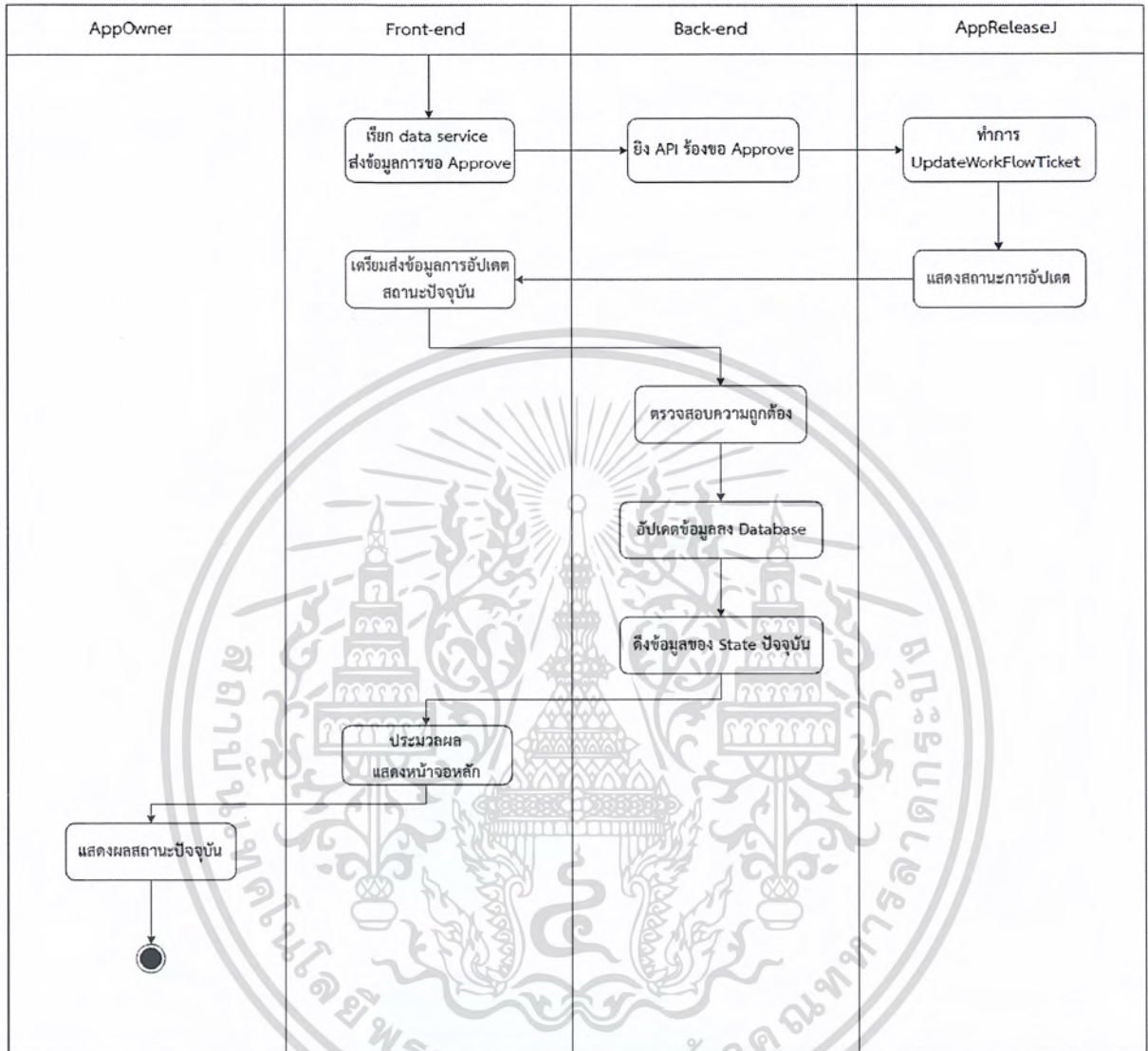
รูปที่ 3.7 Activity Diagram ของหน้าเว็บ Signed off การร้องขอ Release

จากรูปที่ 3.7 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชันให้ QA และ AppOwner สามารถเลือกปุ่ม sign off โดยจะมีการตรวจสอบสิทธิ์ของผู้ใช้งานก่อนจะแสดงปุ่มดังกล่าว เมื่อเลือกกดที่ปุ่ม sign off ระบบจะทำการส่งข้อมูลการขอ sign off ไปยังฝั่ง back-end จากนั้นฝั่ง back-end ทำการยิง API ไปยัง AppreleaseJ เพื่อทำการร้องขอ release แอปพลิเคชันดังกล่าว ( AddNewRule ) เมื่อการร้องขอสำเร็จ ระบบAppreleaseJ จะส่ง response กลับมาเป็นเลข ticket ระบบจะทำการเก็บเลข ticket ดังกล่าวมาเก็บไว้ และทำการอัปเดตข้อมูล ticket รวมทั้งข้อมูลสถานะใหม่ลงฐานข้อมูล จากนั้นระบบจะทำการประมวลผลหน้าเว็บไซต์ และทำการแสดงผลข้อมูลใหม่ให้ผู้ใช้งานได้ทราบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

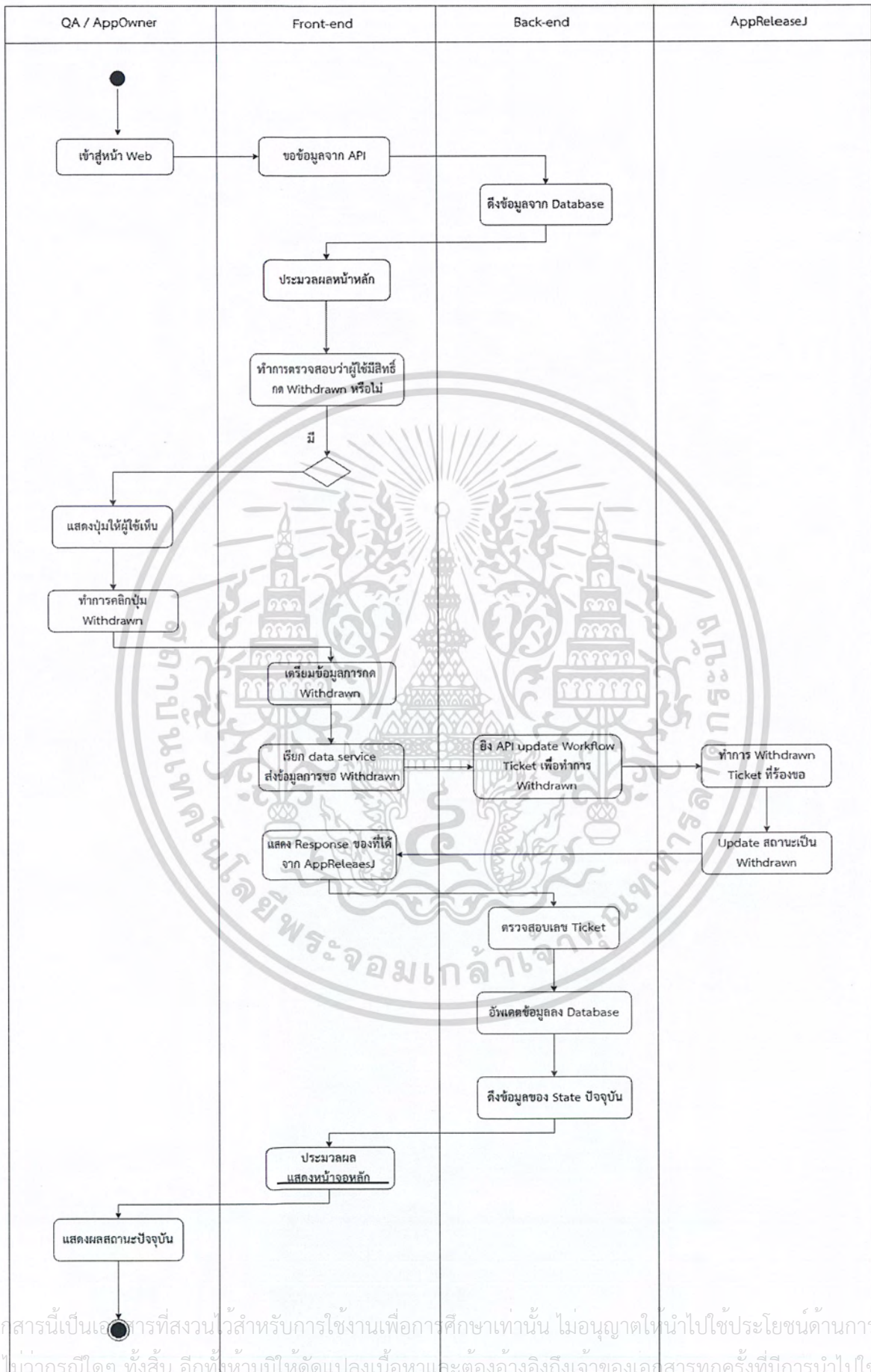
(ต่อ)



รูปที่ 3.8 Activity Diagram ของหน้าเว็บ Approved การร้องขอ Release

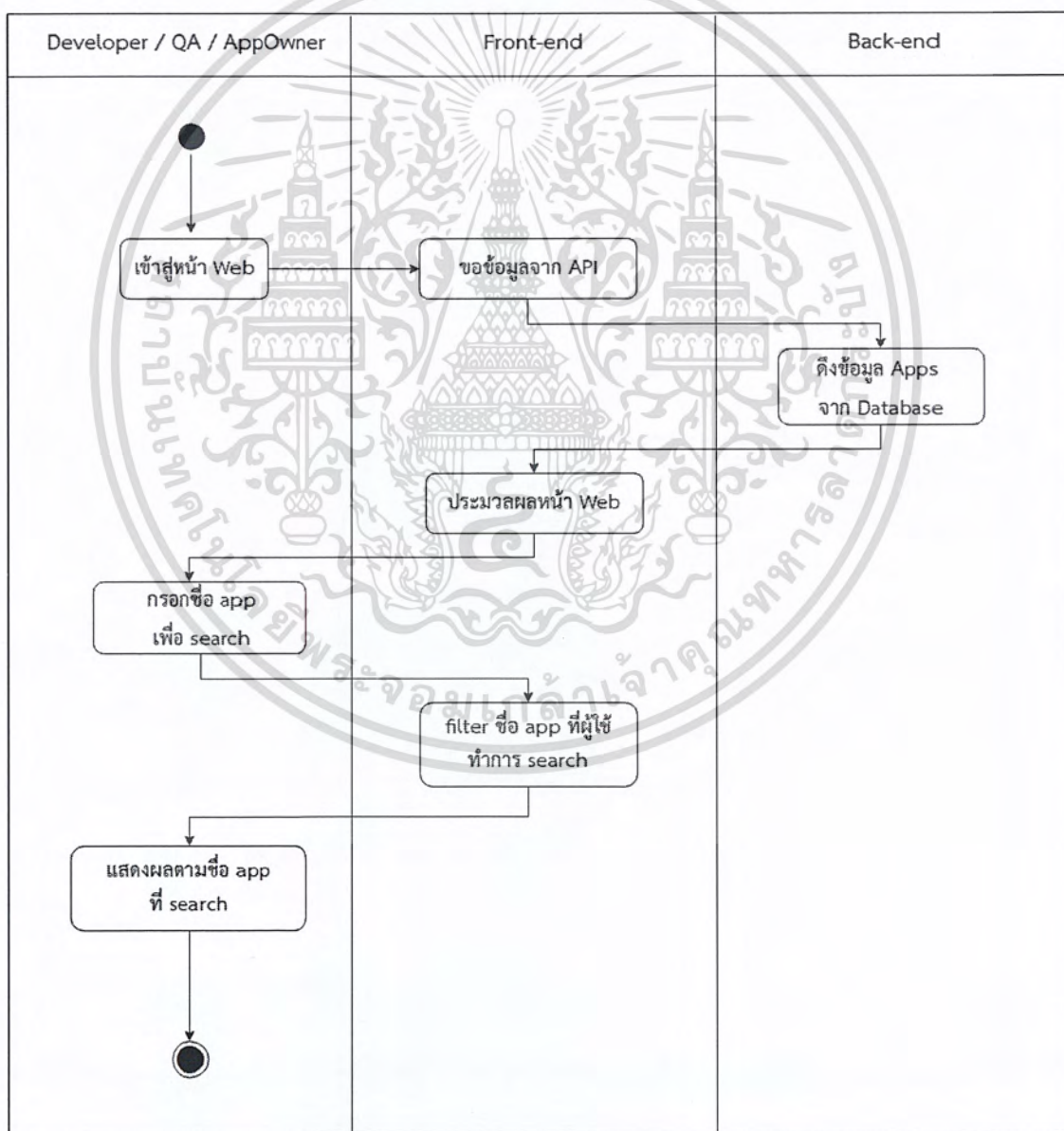
จากรูปที่ 3.8 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชันให้ AppOwner สามารถเลือกปุ่ม Approve โดยจะมีการตรวจสอบสิทธิ์ของผู้ใช้งานก่อนจะแสดงปุ่มดังกล่าว เมื่อเลือกกดที่ปุ่ม Approve ระบบจะทำการส่งข้อมูลการ Approve ไปยังฝั่ง back-end จากนั้นฝั่ง back-end ทำการยิง API ไปยัง AppreleaseJ เพื่อทำการอนุมัติแอปพลิเคชันดังกล่าว (UpdateWorkFlowTicket) เมื่อการอนุมัติสำเร็จ ระบบ AppreleaseJ จะส่ง response การอัปเดตกลับมาเป็น true หลังจากนั้นระบบจะทำการอัปเดตข้อมูลสถานะใหม่ลงฐานข้อมูล

เมื่ออัปเดตสำเร็จ ระบบจะทำการประมวลผลหน้าเว็บไซต์ และทำการแสดงข้อมูลใหม่ให้ผู้ใช้งาน  
 ไม่ได้ทราบ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



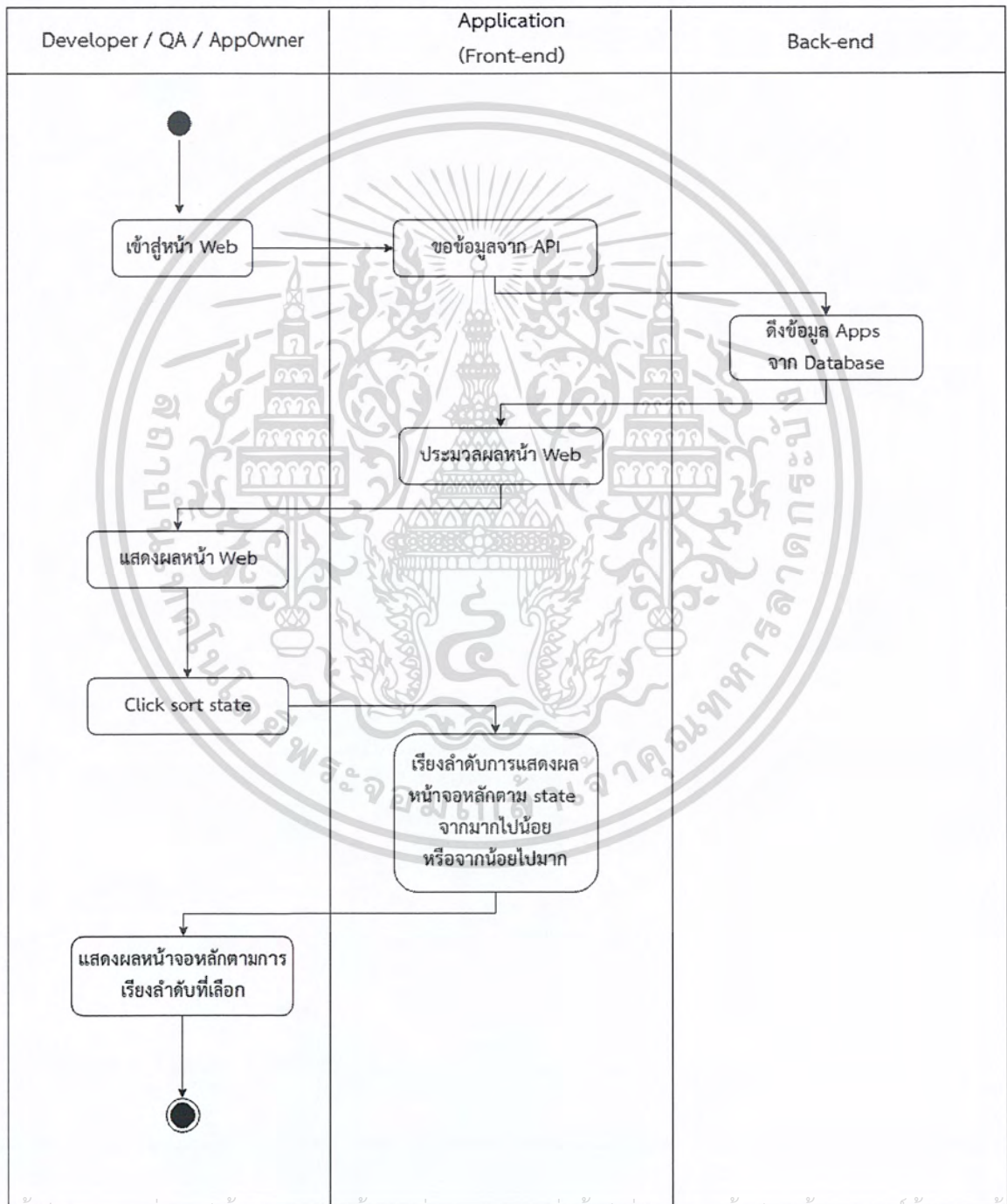
รูปที่ 3.9 Activity Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release

จากรูปที่ 3.9 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชันให้ QA และ AppOwner สามารถเลือกปุ่ม Withdrawn โดยจะมีการตรวจสอบสิทธิ์ของผู้ใช้งานก่อนจะแสดงปุ่มดังกล่าว เมื่อเลือกกดที่ปุ่ม Withdrawn ระบบจะทำการส่งข้อมูลการ Withdrawn ไปยังฝั่ง back-end จากนั้นฝั่ง back-end ทำการยิง API ไปยัง AppreleaseJ เพื่อทำการ Withdrawn แอปพลิเคชันดังกล่าว (UpdateWorkFlowTicket to Withdrawn) เมื่อการ Withdrawn สำเร็จ ระบบAppreleaseJ จะส่ง response การอัปเดตกลับมาเป็น true หลังจากนั้นระบบฝั่ง back-end จะทำการอัปเดตข้อมูลสถานะใหม่ลงฐานข้อมูล เมื่ออัปเดตสำเร็จ ระบบฝั่ง front-end จะทำการประมวลผลหน้าเว็บไซต์ และทำการแสดงข้อมูลใหม่ให้ผู้ใช้งานได้ทราบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 3.10 Activity Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการ Release ใช้

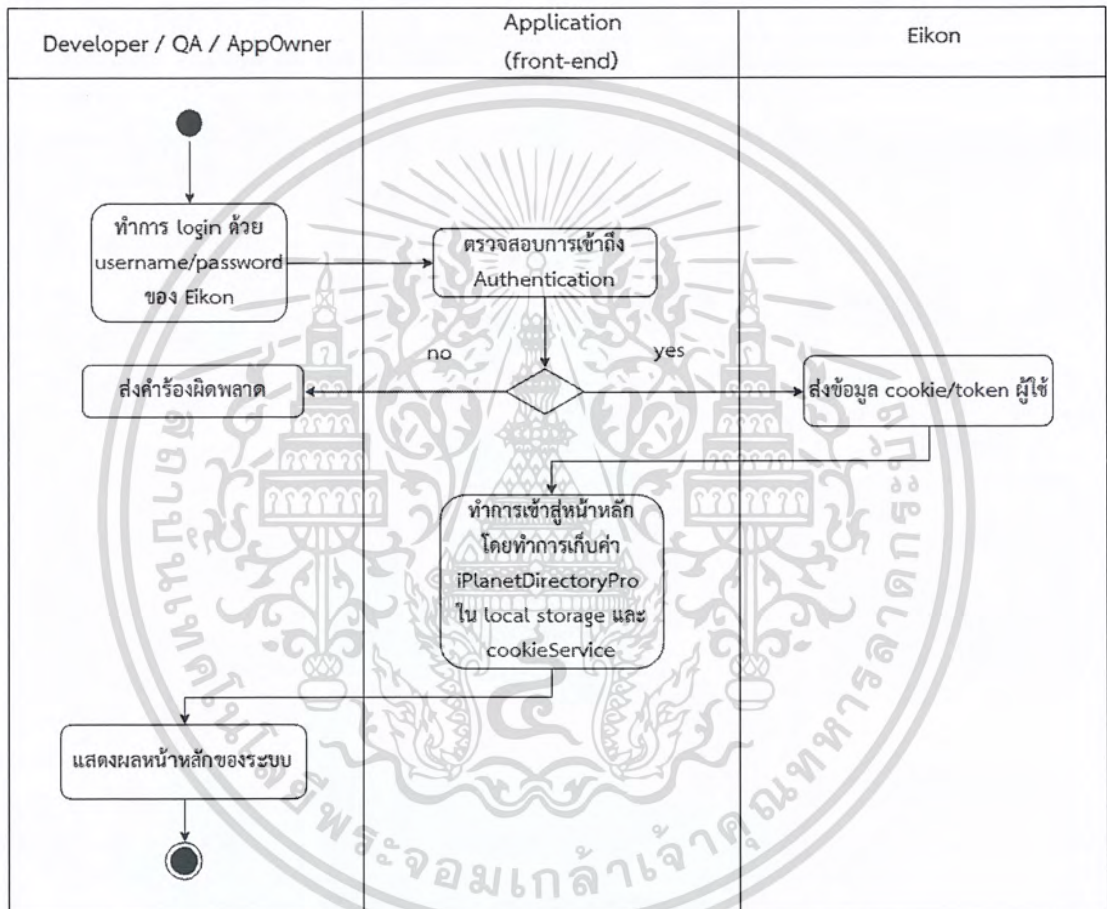
จากรูปที่ 3.10 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชันให้ Developer QA และ AppOwner ผู้ใช้งานสามารถค้นหารายชื่อแอปพลิเคชันที่มีการร้องขอ release ที่ต้องการได้ หลังจากระบบดึงข้อมูลแอปพลิเคชันทั้งหมดจากฐานข้อมูลมาแสดงแล้ว จะมีช่องให้กรอกชื่อแอปพลิเคชันที่ต้องการค้นหา เมื่อผู้ใช้งานกรอกชื่อแอปพลิเคชันดังกล่าว ระบบจะทำการค้นหา รายชื่อดังกล่าว มาแสดงบนหน้าเว็บไซต์ให้ผู้ใช้งานได้ทราบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.11 Activity Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน

จากรูปที่ 3.11 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool จะมีฟังก์ชันให้ Developer QA และ AppOwner ผู้ใช้งานสามารถค้นหารายชื่อแอปพลิเคชันที่มีการร้องขอ release ที่ต้องการได้ หลังจากระบบดึงข้อมูลแอปพลิเคชันทั้งหมดจากฐานข้อมูลมาแสดงแล้ว จะมีปุ่ม sort by state ให้สามารถกดได้ เมื่อเลือกที่ปุ่มดังกล่าว ระบบจะทำการเรียงลำดับการแสดงผลตามสถานะของแต่ละแอปพลิเคชัน และแสดงบนหน้าเว็บไซต์ให้ผู้ใช้งานได้ทราบ



รูปที่ 3.12 Activity Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ

จากรูป 3.12 เมื่อเข้าสู่หน้าเว็บแอปพลิเคชัน AppreleaseTool มีฟังก์ชัน Login ให้ผู้ใช้งานสามารถเลือก โดยเมื่อเลือกที่ปุ่ม Login จะมีหน้าแบบฟอร์มให้กรอก username และ password ที่ใช้ Login Eikon หลังจากเลือกปุ่ม Login ระบบจะทำการตรวจสอบว่าผู้ใช้งานเป็นสมาชิกของ Eikon หรือไม่ ถ้าไม่ระบบจะขึ้นแจ้งเตือนข้อผิดพลาด ถ้าใช่ระบบจากส่งข้อมูล Cookie จาก Eikon มาเก็บไว้ โดยจะทำการเก็บค่า iPlanetDirectoryPro ไว้ใน local storage และ ในCookieService จากนั้นจะ redirect ไปยังหน้าหลักของเว็บไซต์ให้ผู้ใช้งานได้ทราบ

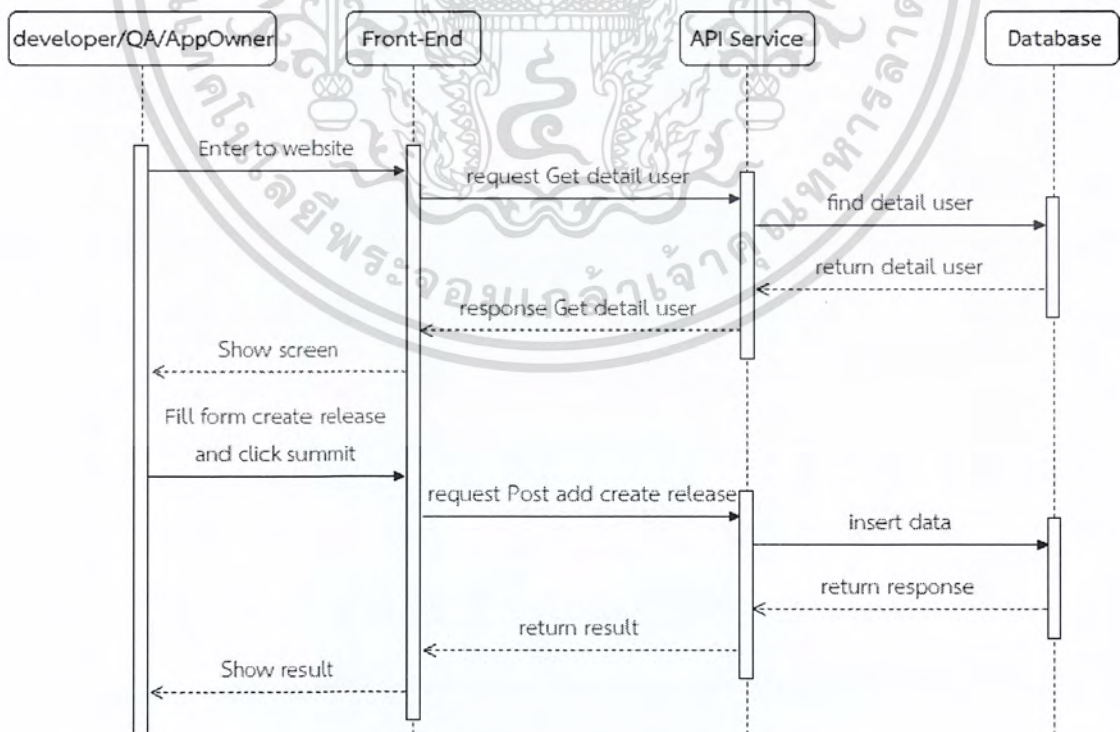
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.5 Sequence Diagram

การออกแบบขั้นตอนการทำงานของระบบเว็บแอปพลิเคชันเป็นลำดับขั้น ในที่นี้ได้กล่าวถึงส่วนการทำงานระหว่างผู้ใช้ (Developer/QA/AppOwner) กับระบบเว็บแอปพลิเคชัน และระบบที่เกี่ยวข้อง โดยมี Sequence Diagram ของแอปพลิเคชัน

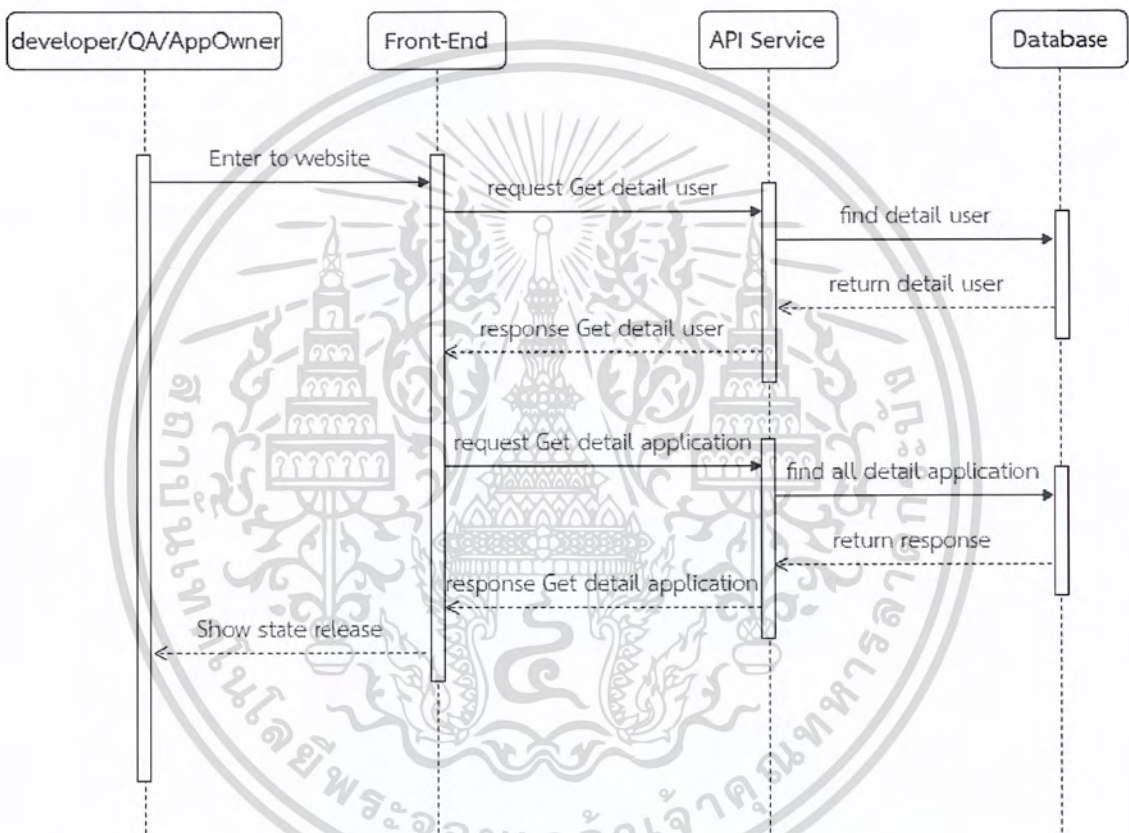
- Sequence Diagram ของหน้าเว็บ Create Release ดังรูป 3.13
- Sequence Diagram ของหน้าเว็บ View State Release ดังรูป 3.14
- Sequence Diagram ของหน้าเว็บ View Detail Release ดังรูป 3.15
- Sequence Diagram ของหน้าเว็บ Signed off การร้องขอ Release ดังรูป 3.16
- Sequence Diagram ของหน้าเว็บ Approved การร้องขอ Release ดังรูป 3.17
- Sequence Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release ดังรูป 3.18
- Sequence Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการร้องขอ Release ดังรูป 3.19
- Sequence Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน ดังรูป 3.20
- Sequence Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ ดังรูป 3.21



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13 Sequence Diagram ของหน้าเว็บ Create Release

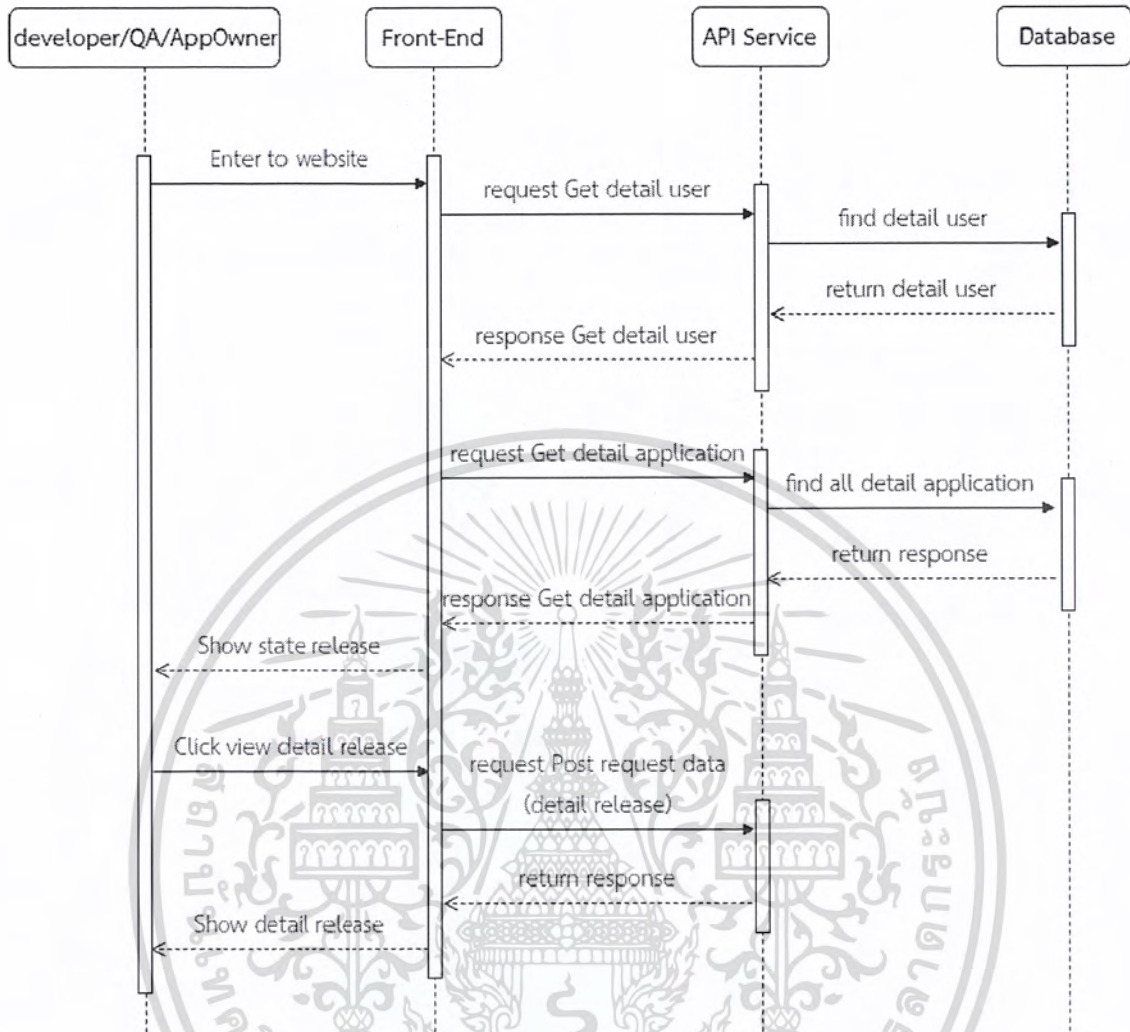
จากรูปที่ 3.13 อธิบายการทำงานในการกรอกแบบฟอร์มหน้าเว็บไซต์การ Create Release โดยจะทำการสร้างรายชื่อแอปพลิเคชันที่ต้องการ release เมื่อมีการเรียกหน้าเว็บ ระบบจะร้องขอข้อมูลจาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ และแสดงข้อมูลเริ่มต้นของระบบจาก API Service ให้ผู้ใช้งานทราบ หลังจากที่ผู้ใช้งานได้กรอกรายชื่อแอปพลิเคชัน เลขเวอร์ชัน และ release note เรียบร้อยแล้วทำการกดปุ่ม submit ระบบจะทำการส่งข้อมูลไปที่ API Service เพื่อเก็บข้อมูลการ create release ลงฐานข้อมูล เมื่อสำเร็จจึงส่งผลลัพธ์และแสดงผลบนหน้าเว็บไซต์



รูปที่ 3.14 Sequence Diagram ของหน้าเว็บ View State Release

จากรูปที่ 3.14 อธิบายการทำงานในการแสดงข้อมูลสถานะของแอปพลิเคชันทั้งหมดที่มีการ release บนหน้าเว็บไซต์ เมื่อมีการเรียกหน้าเว็บระบบจะร้องขอข้อมูลผู้ใช้จาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ และแสดงรายละเอียดผู้ใช้ จากนั้นระบบจะทำการร้องขอข้อมูลแอปพลิเคชันที่มีการ release จาก API Service โดย API Service ทำการดึงข้อมูลแอปพลิเคชันจากฐานข้อมูล เมื่อสำเร็จจึงส่งผลลัพธ์ข้อมูลของแอปพลิเคชันทั้งหมดที่มีการ release และแสดงผลชื่อ เวอร์ชัน สถานะปัจจุบันบนหน้าเว็บไซต์

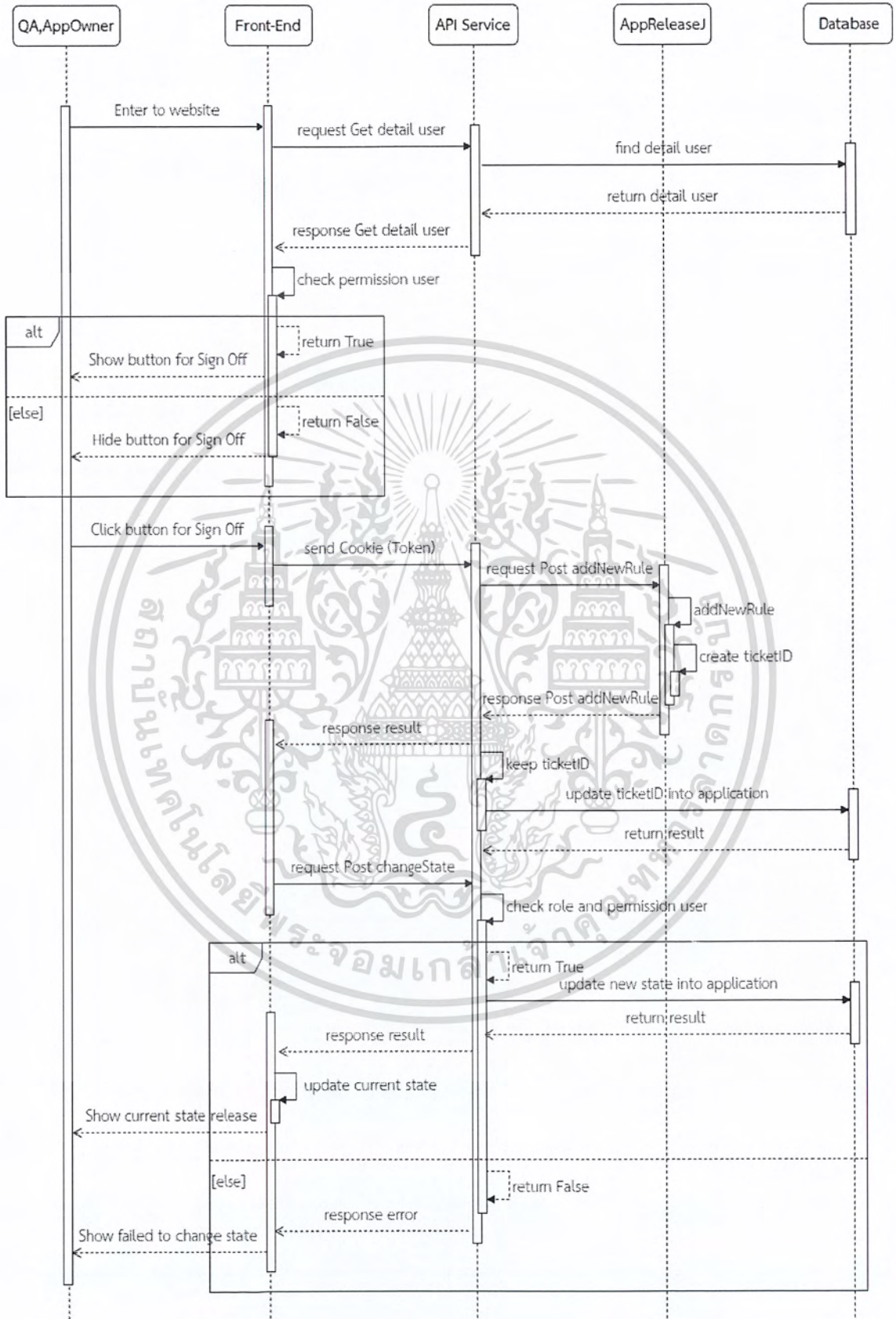
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 Sequence Diagram ของหน้าเว็บ View Detail Release

จากรูปที่ 3.15 อธิบายการทำงานในการแสดงรายละเอียดข้อมูลของแอปพลิเคชันที่มีการ release เมื่อมีการเรียกหน้าเว็บระบบจะร้องขอข้อมูลผู้ใช้จาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ และแสดงรายละเอียดผู้ใช้ จากนั้นระบบจะทำการร้องขอข้อมูลแอปพลิเคชันที่มีการ release จาก API Service โดย API Service ทำการดึงข้อมูลแอปพลิเคชันจากฐานข้อมูล เมื่อสำเร็จจึงส่งผลลัพธ์และแสดงผลบนหน้าเว็บไซต์ เมื่อผู้ใช้เลือกที่ชื่อของแอปพลิเคชันที่ต้องการ ระบบจะทำการร้องขอข้อมูลรายละเอียดของแอปพลิเคชันดังกล่าวจาก API Service เมื่อสำเร็จจึงส่งผลลัพธ์รายละเอียดของแอปพลิเคชันดังกล่าวมาแสดงบนหน้าเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ รูปที่ 3.16 Sequence Diagram ของหน้าเว็บ Signed off การร้องขอ Release นำไปใช้

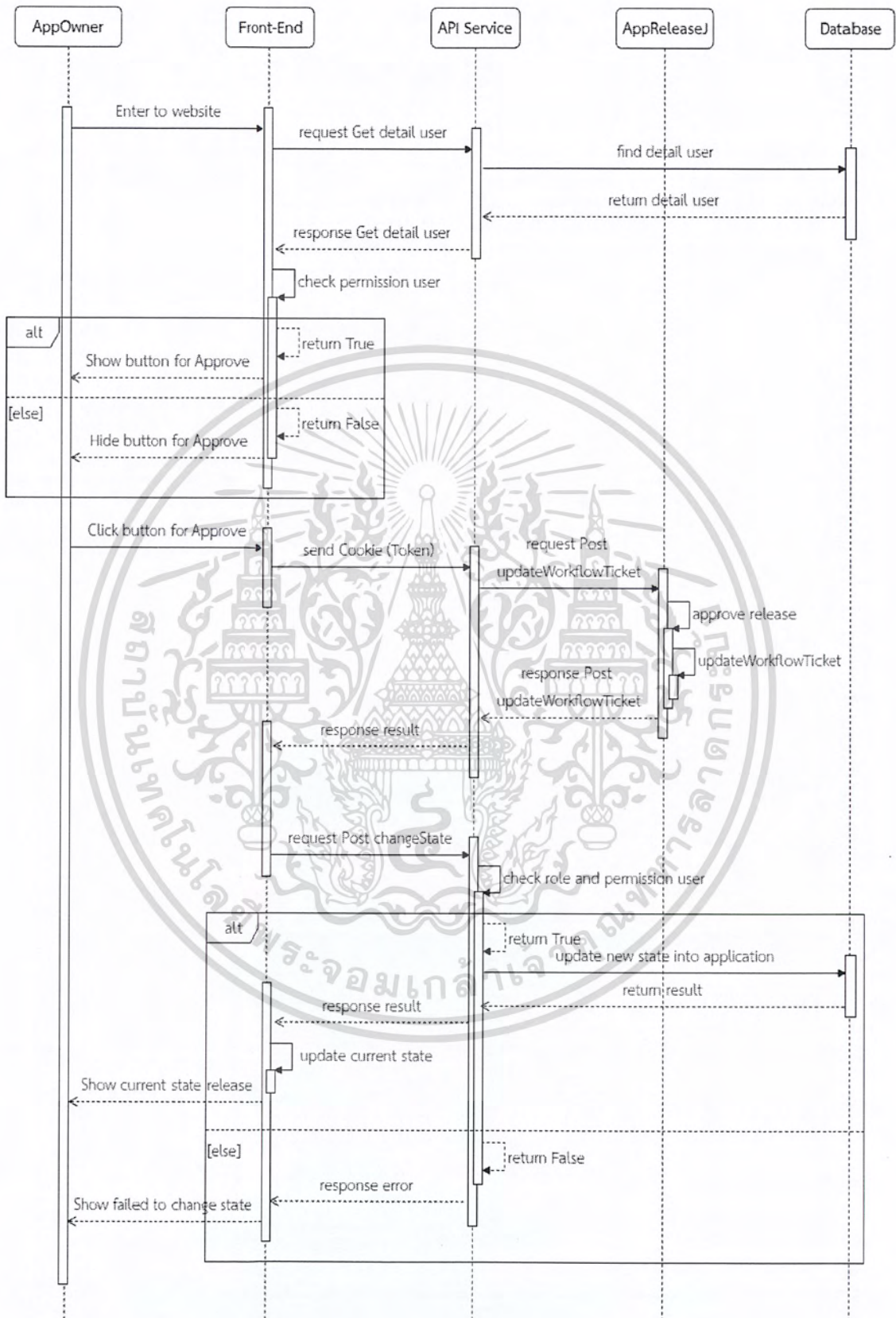
จากรูปที่ 3.16 อธิบายการทำงานในการ signed off แอปพลิเคชันที่มีการ release เมื่อมีการเรียกหน้าเว็บระบบจะร้องขอข้อมูลผู้ใช้จาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ โดยจะมีการดึงข้อมูลผู้ใช้จากฐานข้อมูล และแสดงรายละเอียดผู้ใช้ จากนั้นระบบจะทำการตรวจสอบสิทธิ์ของผู้ใช้ (permission) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการ sign off ระบบจะแสดงปุ่มให้เลือกกด sign off ได้บนหน้าเว็บไซต์
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการ sign off ระบบจะไม่แสดงปุ่ม sign off บนหน้าเว็บไซต์

เมื่อผู้ใช้เลือกที่ปุ่ม sign off ระบบจะดำเนินการ sign off โดยจะส่ง cookie ของผู้ใช้ ไปยัง API Service จากนั้น API Service จะทำการร้องขอ release แอปพลิเคชันดังกล่าวไปยัง everyone โดยส่งการร้องขอไปยัง AppReleaseJ ที่ระบบ AppReleaseJ จะทำการ addNewRule และ create ticketID สำหรับแอปพลิเคชันดังกล่าวที่ทำการ release จากนั้นจึงส่งผลลัพธ์กลับไปยัง API Service และหน้าเว็บไซต์ API Service จะทำการเก็บเลข ticketID ดังกล่าว ไปอัปเดตลงฐานข้อมูลของแอปพลิเคชันนั้นๆ เมื่ออัปเดตเลข ticketID ลงฐานข้อมูลเรียบร้อยแล้ว front-end จะทำการร้องขอการเปลี่ยนสถานะ (changeState) จาก API Service โดย API Service จะทำการตรวจสอบสิทธิ์และบทบาทของผู้ใช้ (permission and role) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการเปลี่ยนสถานะ API Service ทำการเปลี่ยนสถานะ release และทำการเก็บข้อมูลการ changeState release ลงฐานข้อมูล
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการเปลี่ยนสถานะ API Service จะไม่มีการเปลี่ยนสถานะ release

โดยถ้าการเปลี่ยนสถานะสำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ จากนั้นหน้าเว็บจะทำการอัปเดต เพื่อแสดงผลการเปลี่ยนสถานะใหม่ให้ผู้ใช้ได้ทราบ แต่ถ้าการเปลี่ยนสถานะไม่สำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ และแสดงผลการเปลี่ยนสถานะล้มเหลวให้ผู้ใช้ได้ทราบ



เอกสารนี้เป็นเอกสารที่ 3.17 Sequence Diagram ของหน้าเว็บ Approved การร้องขอ Release ไม่เว้นการดำเนินการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

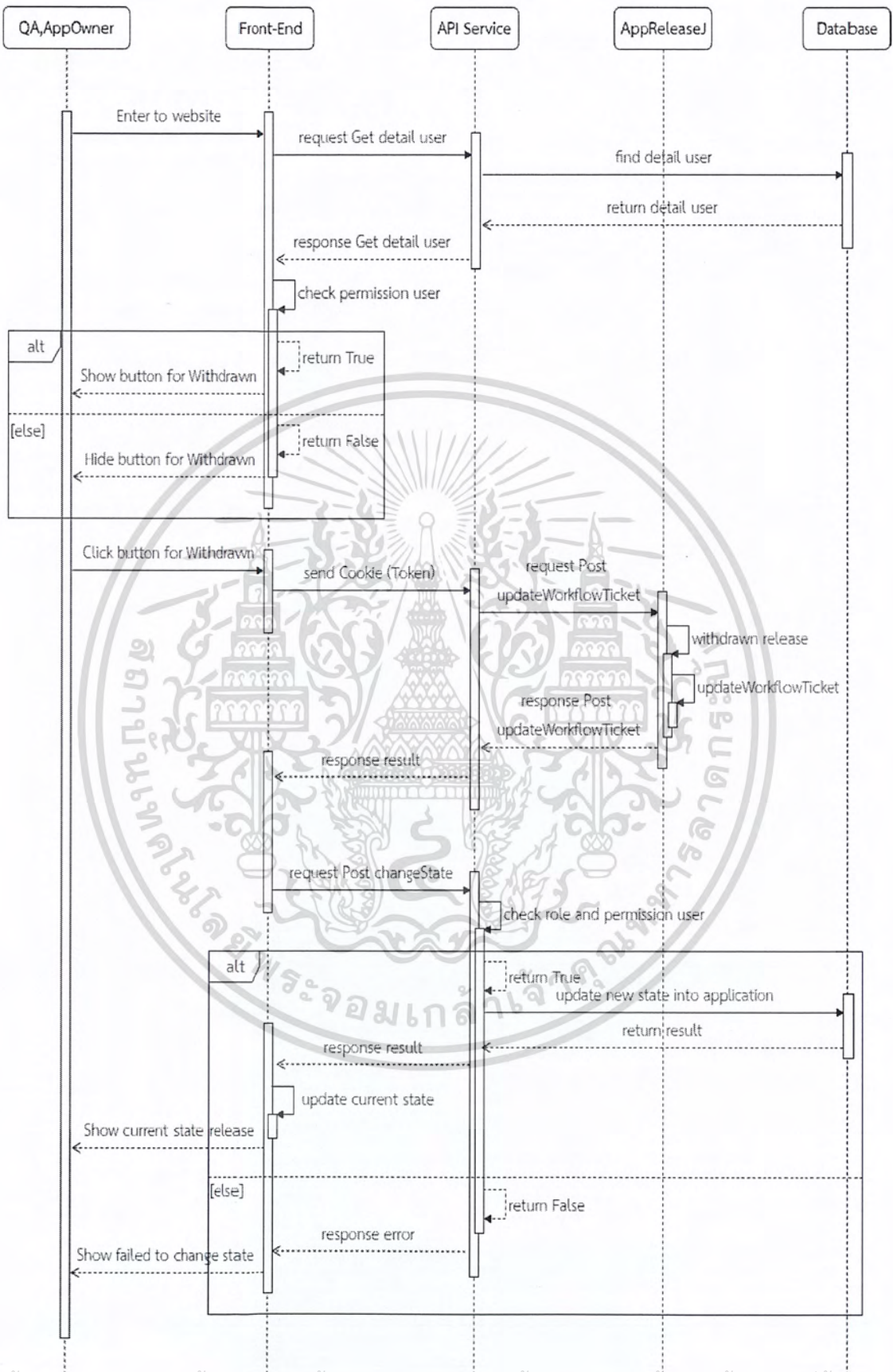
จากรูปที่ 3.17 อธิบายการทำงานในการอนุมัติ (approved) แอปพลิเคชันที่มีการ release สู่ everyone เมื่อมีการเรียกหน้าเว็บระบบจะร้องขอข้อมูลผู้ใช้จาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ โดยจะมีการดึงข้อมูลผู้ใช้จากฐานข้อมูล และแสดงรายละเอียดผู้ใช้ จากนั้นระบบ จะทำการตรวจสอบสิทธิ์ของผู้ใช้ (permission) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการ approve ระบบจะแสดงปุ่มให้เลือกกด approve บนหน้าเว็บไซต์
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการ approve ระบบจะไม่แสดงปุ่ม approve บนหน้าเว็บไซต์

เมื่อผู้ใช้เลือกที่ปุ่ม approve ระบบจะดำเนินการ approve โดยจะส่ง cookie ของผู้ใช้ ไปยัง API Service จากนั้น API Service จะทำการร้องขอการอนุมัติแอปพลิเคชันดังกล่าว ให้สามารถปล่อยไป ยัง everyone ได้ โดยส่งการร้องขออนุมัติไปยัง AppReleaseJ ที่ระบบ AppReleaseJ จะทำการ approve release และ updateWorkflowTicket สำหรับแอปพลิเคชันดังกล่าวที่ทำการ release จากนั้นจึงส่งผลลัพธ์การอัปเดตกลับไปยัง API Service และหน้าเว็บไซต์ เมื่อการอัปเดตสถานะจาก AppReleaseJ สำเร็จ front-end จะทำการร้องขอการเปลี่ยนสถานะ(changeState) จาก API Service โดย API Service จะทำการตรวจสอบสิทธิ์และบทบาทของผู้ใช้ (permission and role) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการเปลี่ยนสถานะ API Service ทำการเปลี่ยนสถานะ release และทำการ เก็บข้อมูลการ changeState release ลงฐานข้อมูล
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการเปลี่ยนสถานะ API Service จะไม่มีการเปลี่ยนสถานะ release

โดยถ้าการเปลี่ยนสถานะสำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ จากนั้นหน้าเว็บจะทำการอัปเดต เพื่อแสดงผลการเปลี่ยนสถานะใหม่ให้ผู้ใช้ได้ทราบ แต่ถ้าการเปลี่ยนสถานะไม่สำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ และแสดงผลการเปลี่ยนสถานะล้มเหลวให้ผู้ใช้ได้ทราบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใด รูปที่ 3.18 Sequence Diagram ของหน้าเว็บ Withdrawn การร้องขอ Release นำไปใช้

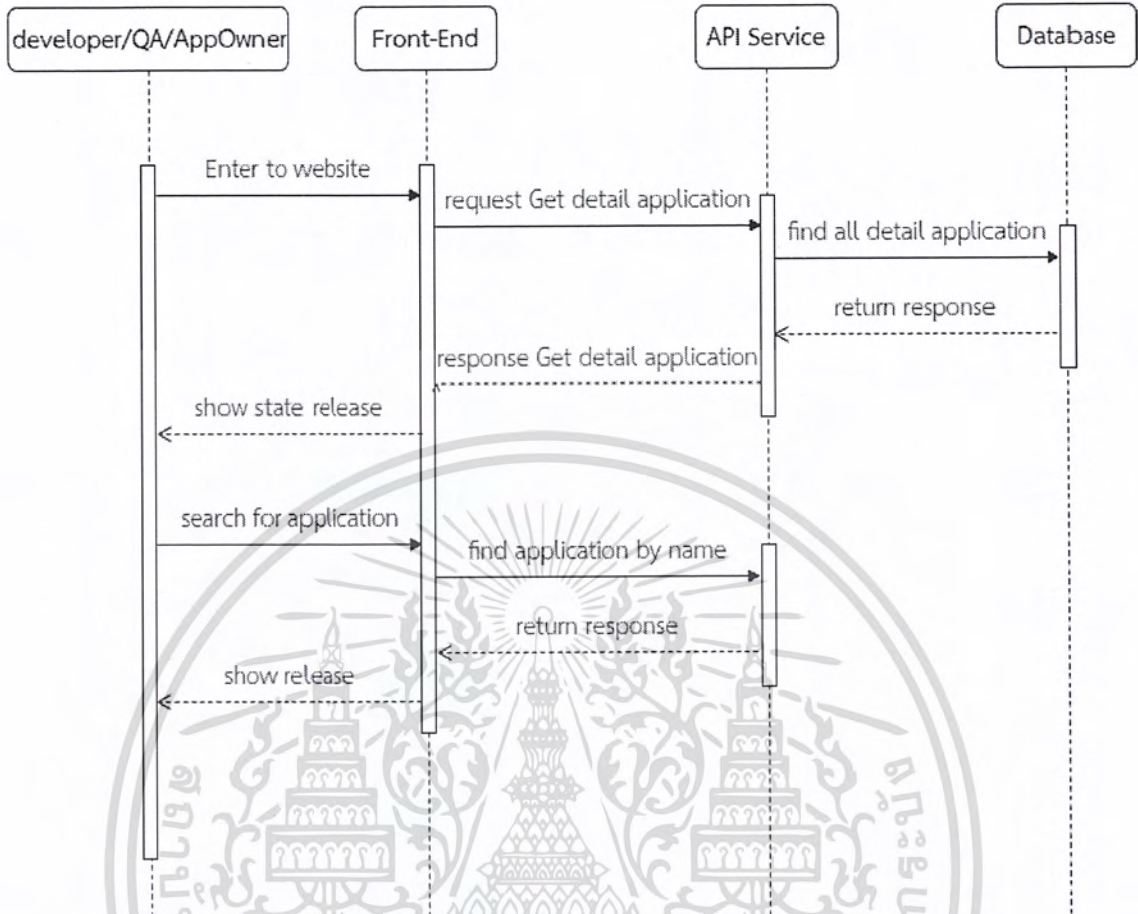
จากรูปที่ 3.18 อธิบายการทำงานในการถอน (withdraw) การ release แอปพลิเคชัน เมื่อมีการเรียกหน้าเว็บระบบจะร้องขอข้อมูลผู้ใช้จาก API Service เพื่อระบุตัวตนผู้ที่เข้าหน้าเว็บไซต์ โดยจะมีการดึงข้อมูลผู้ใช้จากฐานข้อมูล และแสดงรายละเอียดผู้ใช้ จากนั้นระบบจะทำการตรวจสอบสิทธิ์ของผู้ใช้ (permission) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการ withdraw ระบบจะแสดงปุ่มให้เลือกกด withdraw บนหน้าเว็บไซต์ (เฉพาะแอปพลิเคชันที่สถานะปัจจุบันยังไม่ใช่ everyone)
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการ withdraw ระบบจะไม่แสดงปุ่ม withdraw บนหน้าเว็บไซต์

เมื่อผู้ใช้เลือกที่ปุ่ม withdraw ระบบจะดำเนินการ withdraw โดยจะส่ง cookie ของผู้ใช้ ไปยัง API Service จากนั้น API Service จะทำการร้องขอการถอนการ release แอปพลิเคชันดังกล่าว โดยส่งการร้องขอการถอนไปยัง AppReleaseJ ที่ระบบ AppReleaseJ จะทำการ withdrawn release และ updateWorkflowTicket สำหรับแอปพลิเคชันดังกล่าวที่ทำการถอนการ release จากนั้นจึงส่งผลลัพธ์การอัปเดตกลับไปยัง API Service และหน้าเว็บไซต์ เมื่อการอัปเดตสถานะจาก AppReleaseJ สำเร็จ front-end จะทำการร้องขอการเปลี่ยนสถานะ(changeState) จาก API Service โดย API Service จะทำการตรวจสอบสิทธิ์และบทบาทของผู้ใช้ (permission and role) โดยจะทำการตรวจสอบเงื่อนไขดังนี้

- ถ้าผู้ใช้มีสิทธิ์ในการเปลี่ยนสถานะ API Service ทำการเปลี่ยนสถานะ release และทำการเก็บข้อมูลการ changeState release ลงฐานข้อมูล
- ถ้าผู้ใช้ไม่มีสิทธิ์ในการเปลี่ยนสถานะ API Service จะไม่มีการเปลี่ยนสถานะ release

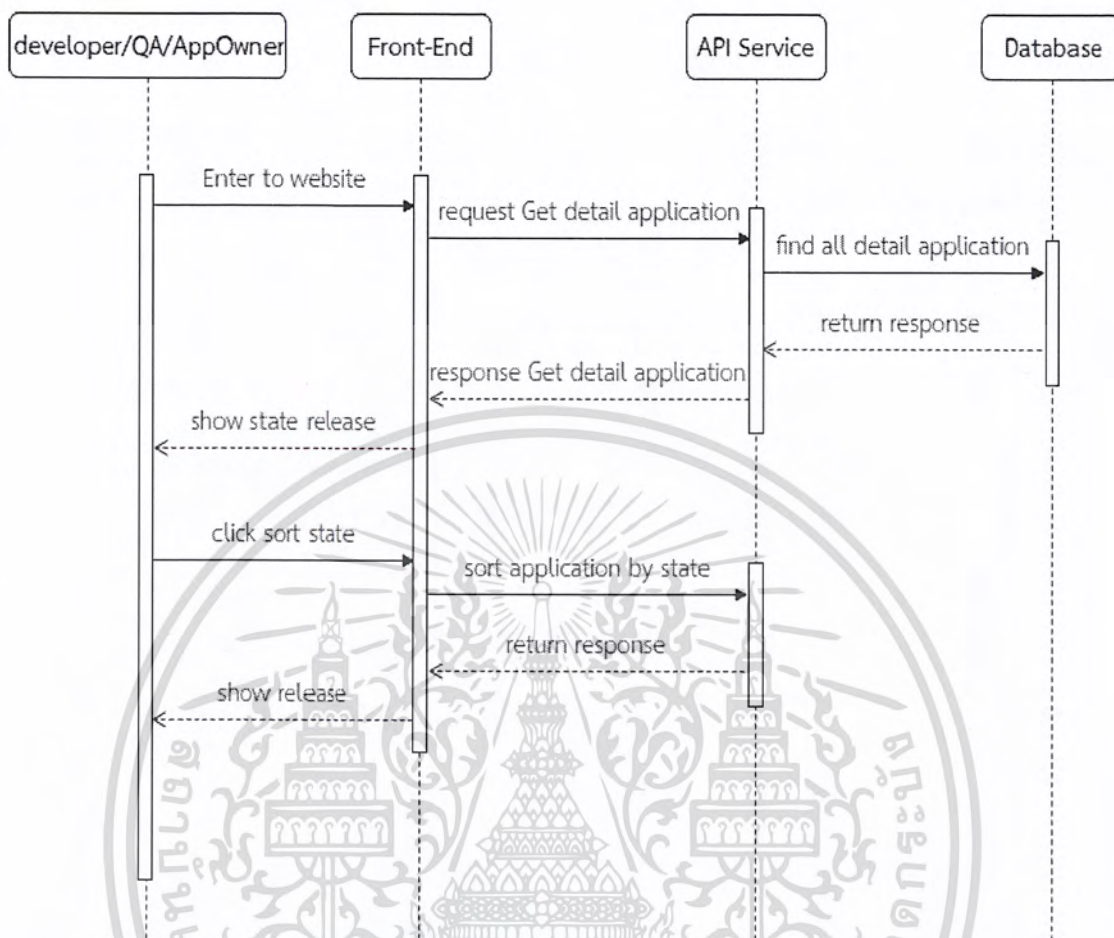
โดยถ้าการเปลี่ยนสถานะสำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ จากนั้นหน้าเว็บจะทำการอัปเดต เพื่อแสดงผลการเปลี่ยนสถานะใหม่ให้ผู้ใช้ได้ทราบ แต่ถ้าการเปลี่ยนสถานะไม่สำเร็จ ระบบส่งผลลัพธ์คืนกลับมายังหน้าเว็บไซต์ และแสดงผลการเปลี่ยนสถานะล้มเหลวให้ผู้ใช้ได้ทราบ



รูปที่ 3.19 Sequence Diagram ของหน้าเว็บสำหรับการค้นหาชื่อแอปพลิเคชันที่มีการ Release

จากรูปที่ 3.19 อธิบายการทำงานในการค้นหาชื่อแอปพลิเคชันที่มีการร้องขอ Release เมื่อมีการเรียกหน้าเว็บระบบจะทำการร้องขอข้อมูลแอปพลิเคชันทั้งหมดที่มีการ release จาก API Service โดย API Service ทำการดึงข้อมูลแอปพลิเคชันทั้งหมดจากฐานข้อมูล เมื่อสำเร็จจึงส่งผลลัพธ์กลับและแสดงผลบนหน้าเว็บไซต์ เมื่อผู้ใช้กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหาลงในช่องค้นหา ระบบจะทำคัดกรอกรายชื่อและข้อมูลรายละเอียดของแอปพลิเคชันดังกล่าวจาก API Service เมื่อสำเร็จจึงส่งผลลัพธ์รายละเอียดของแอปพลิเคชันดังกล่าวมาแสดงผลบนหน้าเว็บไซต์

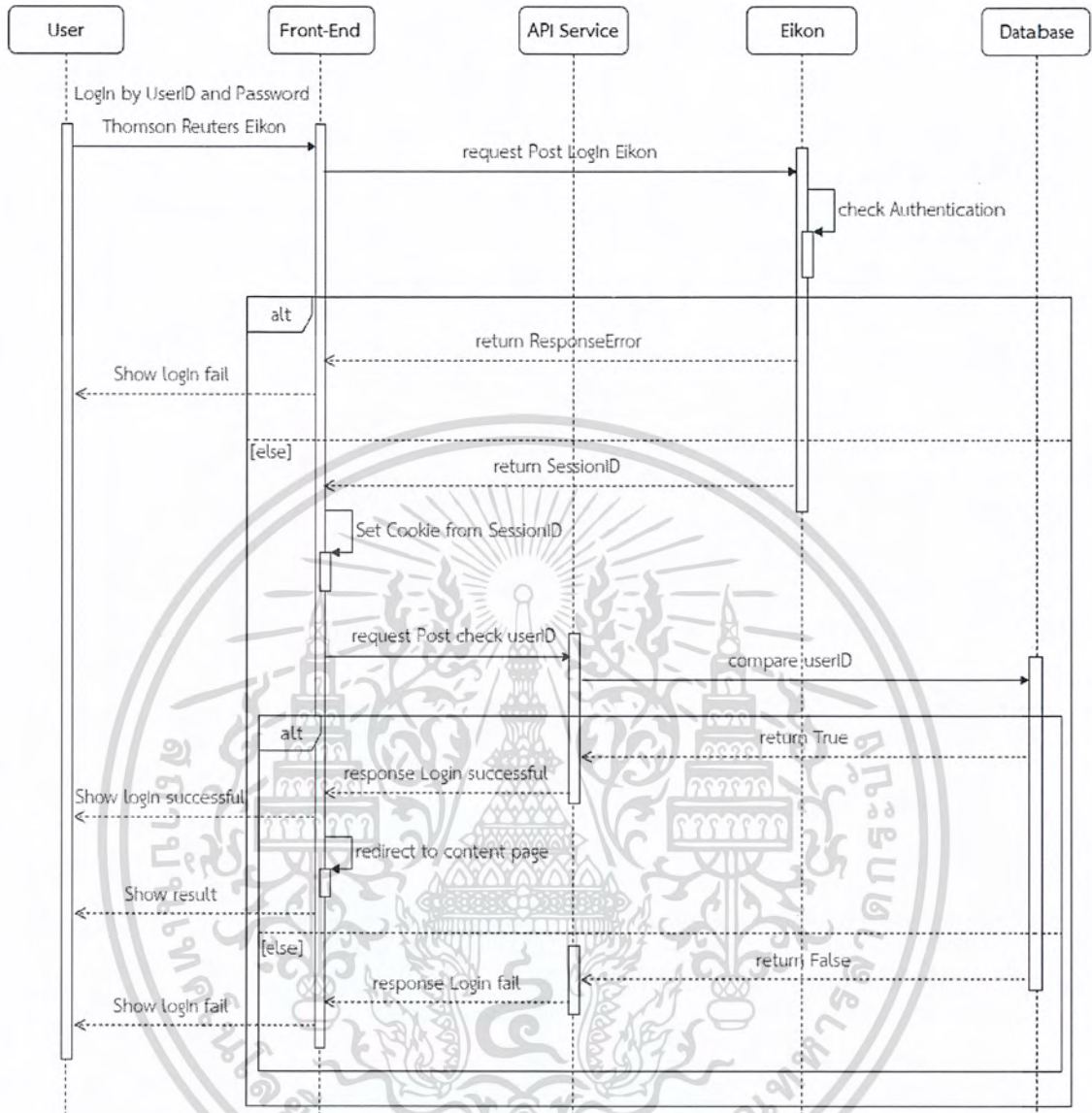
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 Sequence Diagram ของหน้าเว็บสำหรับการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน

จากรูปที่ 3.20 อธิบายการทำงานในการการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชัน เมื่อมีการเรียกหน้าเว็บระบบจะทำการร้องขอข้อมูลแอปพลิเคชันทั้งหมดที่มีการ release จาก API Service โดย API Service ทำการดึงข้อมูลแอปพลิเคชันทั้งหมดจากฐานข้อมูล เมื่อสำเร็จจึงส่งผลลัพธ์กลับและแสดงผลบนหน้าเว็บไซต์ เมื่อผู้ใช้เลือกที่ปุ่ม sort by state ระบบจะทำการเรียงลำดับการแสดงผลหน้าเว็บไซต์ตามสถานะของแต่ละแอปพลิเคชัน เมื่อสำเร็จเรียงลำดับสำเร็จจึงส่งผลลัพธ์มาแสดงผลบนหน้าเว็บไซต์ โดยถ้าเริ่มต้นระบบเรียงลำดับจากมากไปน้อย เมื่อผู้ใช้งานเลือกที่ปุ่ม sort by state อีกครั้ง ระบบจะทำการเรียงลำดับตามสถานะของแอปพลิเคชันจากน้อยไปมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 Sequence Diagram ของหน้าเว็บสำหรับ login เข้าสู่ระบบ

จากรูปที่ 3.21 อธิบายการทำงานในการ login เข้าสู่ระบบ เมื่อผู้ใช้กรอก userID และ password ของ Eikon แล้วทำการกดปุ่ม login ระบบจะทำการร้องขอการ login Eikon ไปยัง API Eikon จากนั้น Eikon ทำการตรวจสอบ authentication ถ้า userID หรือ password ผิด ระบบจะส่ง response Error กลับมายังระบบ และระบบจะแสดงข้อความ login fail บนหน้า login ถ้า userID หรือ password ถูก ระบบจะส่ง sessionID กลับมายังระบบ และระบบจะทำการเก็บ cookie จาก sessionID จากนั้นระบบจะทำการร้องขอตรวจสอบ userID จาก API service โดย API service

เปรียบเทียบ userID ในฐานข้อมูล ถ้า userID ตรงกัน return true แสดงข้อความ login successful และ redirect ไปยังหน้าหลักของเว็บไซต์ ถ้า userID ไม่ตรงกัน return false และแสดงข้อความ login fail บนหน้า login

### 3.2.6 การออกแบบฐานข้อมูล

ในการออกแบบฐานข้อมูลของระบบ จะทำการเก็บข้อมูลโดยใช้ MongoDB เป็นฐานข้อมูลแบบ NoSQL โดยเก็บข้อมูลเป็นแบบ JSON ( JavaScript Object Notation ) มีโครงสร้างในการจัดเก็บข้อมูลเป็นแบบ key และ value ซึ่งจะเก็บข้อมูลที่สำคัญคือ ข้อมูลแอปพลิเคชันที่ต้องการ release

#### ข้อมูล application

โครงสร้างการเก็บข้อมูลของ collection application เริ่มต้นจะประกอบไปด้วย \_id releaseID appName appVersion currentState และ history โดย history จะมีหน้าที่เก็บ state ของ application ซึ่ง state ที่ 1 (wait for QA test) จะถูกจัดเก็บเมื่อมีการ create release จากหน้า frontend ประกอบด้วย state date createBy และ releaseNote

```
{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 1,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593248.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uuid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    }
  ]
}
```

รูปที่ 3.22 โครงสร้างการเก็บข้อมูล application หลังจากการ create release

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ QA ทำการกด sign off จากหน้า frontend ระบบจะทำการเก็บ state ที่ 2 (wait for AppOwner Approve) และ ticketID ที่ส่งมาจากระบบ AppReleaseJ ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ประกอบด้วย state date signedoffBy และ releaseNote

```

{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 2,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593248.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uuid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    },
    {
      "state" : 2,
      "date" : 15411147526167.0,
      "signedoffBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    }
  ],
  "ticketID" : "10017601"
}

```

รูปที่ 3.23 โครงสร้างการเก็บข้อมูล application หลังจากการกด signed off

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ AppOwner ทำการกด approved จากหน้า frontend ระบบจะทำการเก็บ state ที่ 3 (wait for TR Internal) ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ซึ่งประกอบด้วย state date และ approvedBy

```

{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 3,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593248.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    },
    {
      "state" : 2,
      "date" : 1541147526167.0,
      "signedoffandrequestBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uid" : "003"
      }
    },
    {
      "state" : 3,
      "date" : 1542076086619.0,
      "approvedBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uid" : "003"
      }
    }
  ],
  "ticketID" : "10017601"
}

```

รูปที่ 3.24 โครงสร้างการเก็บข้อมูล application หลังการกด approved

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจาก state เปลี่ยนเป็น wait for TR Internal ระบบจะทำการ check สถานะจริงจาก AppReleaseJ หากตรวจสอบแล้วตรงกับสถานะ Internal Release Complete - All DCs ระบบจะทำการเปลี่ยน state เป็น Internal และจะทำการเก็บ state ที่ 4 (Internal) ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ประกอบด้วย state และ date

```

{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 4,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593248.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    },
    {
      "state" : 2,
      "date" : 1541147526167.0,
      "signedoffandrequestBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uid" : "003"
      }
    },
    {
      "state" : 3,
      "date" : 1542076086619.0,
      "approvedBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uid" : "003"
      }
    },
    {
      "state" : 4,
      "date" : 1542076203017.0
    }
  ],
  "ticketID" : "10017601"
}

```

รูปที่ 3.25 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น internal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจาก state เปลี่ยนเป็น Internal ระบบจะทำการร้องขอ AppReleaseJ ให้อนุมัติไปยัง everyone โดยอัตโนมัติ หลังจากระบบ AppReleaseJ อนุมัติและเปลี่ยนสถานะเรียบร้อยแล้ว ระบบจะทำการ check สถานะจริงจาก AppReleaseJ AppReleaseJ หากตรวจสอบแล้วว่าตรงกับสถานะ Everyone Release Ops Approval – primary DC (OpsApprOneDCExt) ระบบจะทำการเปลี่ยน state เป็น wait for Everyone และจะทำการเก็บ state ที่ 5 (wait for Everyone) ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ประกอบด้วย state และ date

```
{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 5,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593248.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uuid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    },
    {
      "state" : 2,
      "date" : 1541147526167.0,
      "signedoffandrequestBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    },
    {
      "state" : 3,
      "date" : 1542076086619.0,
      "approvedBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    },
    {
      "state" : 4,
      "date" : 1542076203017.0
    },
    {
      "state" : 5,
      "date" : 1542176204812.0
    }
  ],
  "ticketID" : "10017601"
}
```

เอกสารนี้เป็นเอกสารทบทวนวิศวกรรมเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้โดยไม่ได้รับอนุญาตจะถือว่าผิดกฎหมาย

ไม่ว่ากรณีใดๆ ขอสงวนสิทธิ์ในสิ่งที่ปรากฏและไม่รับผิดชอบต่อการใช้งานเอกสารฉบับนี้

รูปที่ 3.26 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น wait for Everyone

หลังจาก state เปลี่ยนเป็น wait for Everyone ระบบจะทำการ check สถานะจริงจาก AppReleaseJ หากตรวจสอบพบว่าแอปพลิเคชันดังกล่าว release สู่ everyone เรียบร้อยแล้ว ระบบจะทำการเปลี่ยน state เป็น Everyone และจะทำการเก็บ state ที่ 6 (Everyone) ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ประกอบด้วย state และ date

```

{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 6,
  "history" : [
    {
      "state" : 1,
      "date" : 1541057593246.0,
      "createBy" : {
        "firstName" : "Siriporn",
        "lastName" : "Kiwatthana",
        "uuid" : "001"
      },
      "releaseNote" : "Hi, I am releaseA!!"
    },
    {
      "state" : 2,
      "date" : 1541147526167.0,
      "signedoffandrequestBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    },
    {
      "state" : 3,
      "date" : 1542076086619.0,
      "approvedBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    },
    {
      "state" : 4,
      "date" : 1542076203017.0
    },
    {
      "state" : 5,
      "date" : 1542176204812.0
    },
    {
      "state" : 6,
      "date" : 1545077206610.0
    }
  ],
  "ticketID" : "10017601"
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใ้ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
**รูปที่ 3.27** โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น Everyone  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ระบบทำการกด withdrawn จากหน้า frontend ระบบจะทำการเก็บ state เป็น 0 (withdraw) ลงในฐานข้อมูล application โดยจะเพิ่ม state ลงไปใน object ของ history ประกอบด้วย state date rejectedBy และ releaseNote

```
{
  "_id" : ObjectId("5bdaabc6003cf7247ce63eb5"),
  "releaseID" : "1",
  "appName" : "releaseA",
  "appVersion" : "1.0",
  "currentState" : 0,
  "ticketID" : "10017601",
  "history" : [
    {
      "state" : 0,
      "date" : 1542080663545.0,
      "releaseNote" : "withdrawn",
      "rejectedBy" : {
        "firstName" : "Tawan",
        "lastName" : "Tananwan",
        "uuid" : "003"
      }
    }
  ]
}
```

รูปที่ 3.28 โครงสร้างการเก็บข้อมูล application หลังจากเปลี่ยนสถานะเป็น Withdrawn

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการดำเนินงาน

บทนี้เป็นการกล่าวถึงผลการสร้างระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ซึ่งเว็บแอปพลิเคชันนี้ได้ถูกพัฒนาตามทีออกแบบไว้ในบทที่ 3 และถูกพัฒนาโดยใช้โปรแกรม Microsoft Visual Studio Code โดยในส่วนหน้าเว็บไซต์เน้นการใช้ภาษา HTML, CSS, TypeScript และ Angular6 ทดสอบผลบนคอมพิวเตอร์พกพาที่ทำงานบนระบบปฏิบัติการ Microsoft Windows 7 และมีการ deploy บนเครื่อง Server ของบริษัท ซึ่ง domain ที่ใช้เข้าถึงเว็บจะเป็น eikonci.int.thomsonreuters.com หรือ 172.20.199.167 ซึ่งเว็บดังกล่าวจำเป็นต้องเข้าผ่านเครือข่ายระบบไร้สายของบริษัทเท่านั้นจึงจะใช้งานได้

ในการสร้างระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ระบบสามารถทำรายการได้ดังต่อไปนี้

- 1) ทำการสร้างรายชื่อแอปพลิเคชันที่ต้องการร้องขอ release ( Create release )
- 2) ทำการ Signed off แอปพลิเคชันที่มีการร้องขอ release ( Signed off release )
- 3) ทำการอนุมัติแอปพลิเคชันที่ต้องการ release ไป internal ( Approved release )
- 4) ทำการยกเลิกการ release แอปพลิเคชัน ( Withdrawn release )
- 5) ทำการดูรายละเอียดแอปพลิเคชันที่มีการ release ไป everyone
- 6) ทำการดูสถานะของแอปพลิเคชันที่มีการ release ไป everyone
- 7) ทำการค้นหารายชื่อแอปพลิเคชันที่มีการ create release
- 8) ทำการเรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชันที่มีการ release
- 9) ทำการตรวจสอบสถานะจริงของแอปพลิเคชันในระบบ AppReleaseJ โดยอัตโนมัติ
- 10) ทำการอนุมัติแอปพลิเคชันที่ต้องการ release ไป everyone โดยอัตโนมัติ
- 11) ทำการลงชื่อเข้าใช้งานระบบ

#### 4.1 ผลลัพธ์การพัฒนาเว็บเอพีไอ (Web API Service) ของระบบ

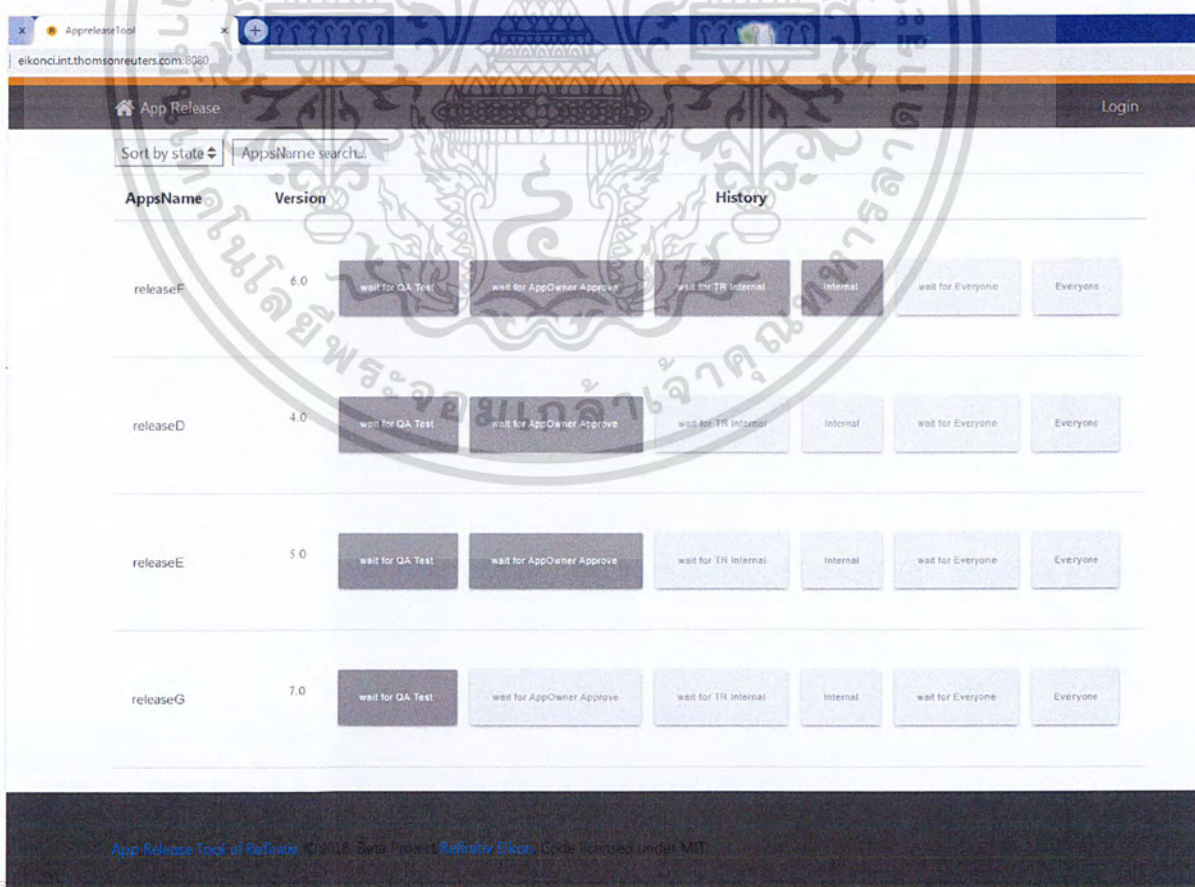
สร้างขึ้นเพื่อใช้ในระบบ Back-End ของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ( AppReleaseTool System ) ในการเก็บข้อมูล เปลี่ยนแปลงข้อมูล ตรวจสอบข้อมูล โดยสร้างหน้าเว็บข้อมูลการใช้งาน และบริการสำหรับ API AppReleaseTool System ผู้จัดทำสร้างขึ้นโดยใช้ NodeJS ในการสร้าง

โดยรายละเอียดของ API AppReleaseTool System ประกอบด้วย API ที่มีในระบบ รายละเอียดของ URL parameter ในการเรียกใช้ API และรายละเอียดการส่ง Request เข้าไปให้ API ทำงาน รวมถึงรายละเอียด Response ของ API ที่ส่งกลับมายังระบบที่เรียกใช้งาน

Response ของ API AppReleaseTool System สามารถส่งข้อมูลมาในรูปแบบของ XML หรือ JSON ซึ่งเป็นรูปแบบมาตรฐานที่ทุกภาษาที่เขียนโปรแกรมทางคอมพิวเตอร์ สามารถนำข้อมูลที่ส่งกลับมาไปใช้งานในระบบได้

## 4.2 ผลลัพธ์การพัฒนาเว็บแอปพลิเคชันของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้น ภายในองค์กรของบริษัท Refinitiv

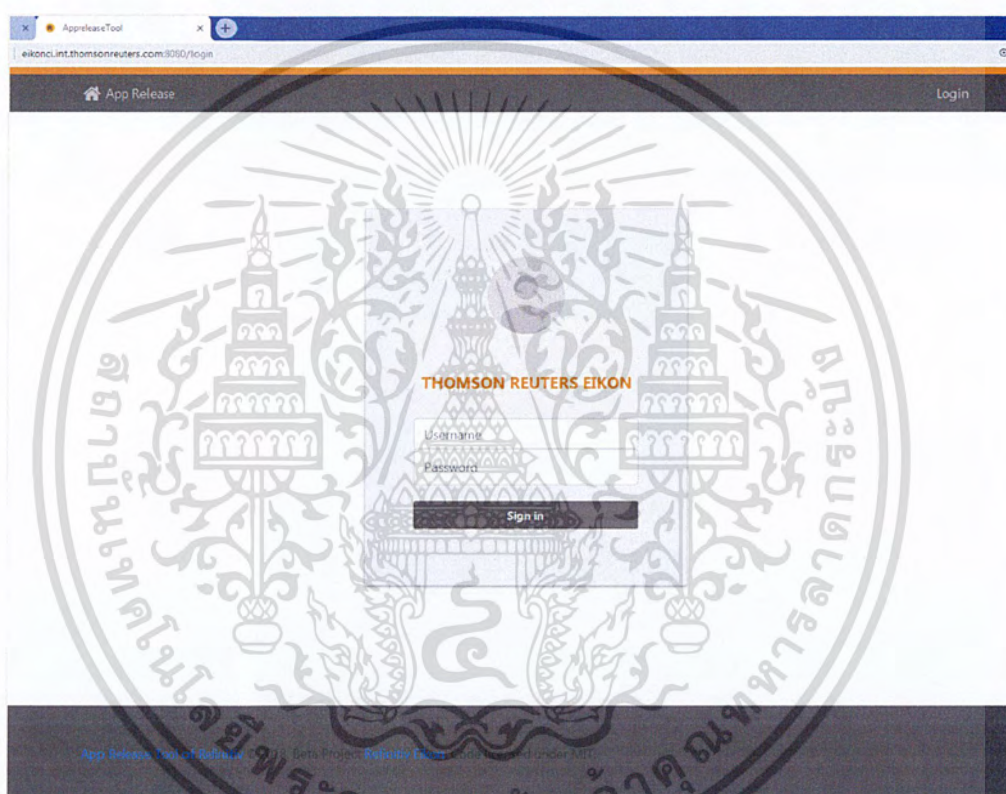
ทางผู้จัดทำได้ทำการสร้าง Back-End ออกแบบ API Service ให้มีการเชื่อมต่อการทำงานกับ Front-End และแบ่งการเขียนโค้ดออกเป็นส่วน เพื่อให้สะดวกต่อการแก้ไขเปลี่ยนแปลงระบบในอนาคต ( เนื้อหาบางส่วนของ การแสดงหน้าเว็บแอปพลิเคชันได้ทำการสมมติข้อมูล และปิดข้อความบางส่วนไว้สำหรับข้อมูลความลับ และเพื่อความปลอดภัยของบริษัท )



เอกสาร

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิใช้รูปที่ 4.1 หน้าแรกของเว็บแอปพลิเคชันเอกสารทุกครั้งที่มีการนำไปใช้

รูป 4.1 เป็นหน้าแรกของเว็บแอปพลิเคชัน ในส่วนของผู้ใช้งาน สามารถดูรายการแอปพลิเคชันที่มีการ release ทั้งหมดได้ โดยจะแสดง ชื่อ เลขเวอร์ชัน และประวัติการแสดงสถานะ มีฟังก์ชัน “Sort by state” เพื่อเรียงลำดับรายการแอปพลิเคชันตามสถานะ มีฟังก์ชัน “AppsName search” เพื่อใช้ทำการค้นหารายชื่อแอปพลิเคชันที่ต้องการ มีฟังก์ชัน Login ให้ผู้ใช้งานลงชื่อเข้าสู่ระบบ เพื่อทำรายการอื่นๆเพิ่มเติม ผู้ใช้สามารถเลือกที่ปุ่มรูปบ้านเพื่อทำการ relode หน้าเว็บแอปพลิเคชันได้ และผู้ใช้สามารถเลือกที่ชื่อของแต่ละแอปพลิเคชัน เพื่อดูรายละเอียดเพิ่มเติมของแอปพลิเคชันได้



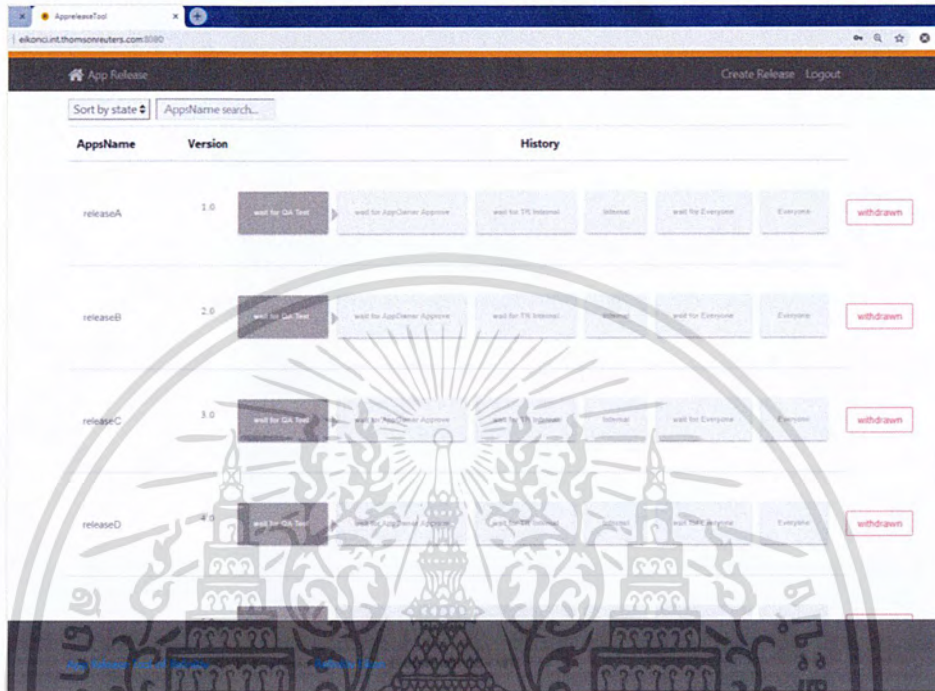
รูปที่ 4.2 หน้าเว็บแสดง Login

รูป 4.2 เป็นหน้า login เข้าสู่ระบบ หลังจากผู้ใช้เลือกที่ปุ่ม login ที่หน้าแรกของแอปพลิเคชัน ระบบจะแสดงหน้าเว็บสำหรับให้ผู้ใช้ลงชื่อเข้าใช้งาน โดยการลงชื่อเข้าใช้งานนั้น ผู้ใช้จำเป็นต้องกรอก username และ password ให้ถูกต้อง ( username และ password จำเป็นต้องเป็นของ Eikon ) เมื่อผู้ใช้กรอกข้อมูลเรียบร้อย ดังรูป 4.3 แล้วกดปุ่ม Sign in ระบบจะดำเนินการ login เข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

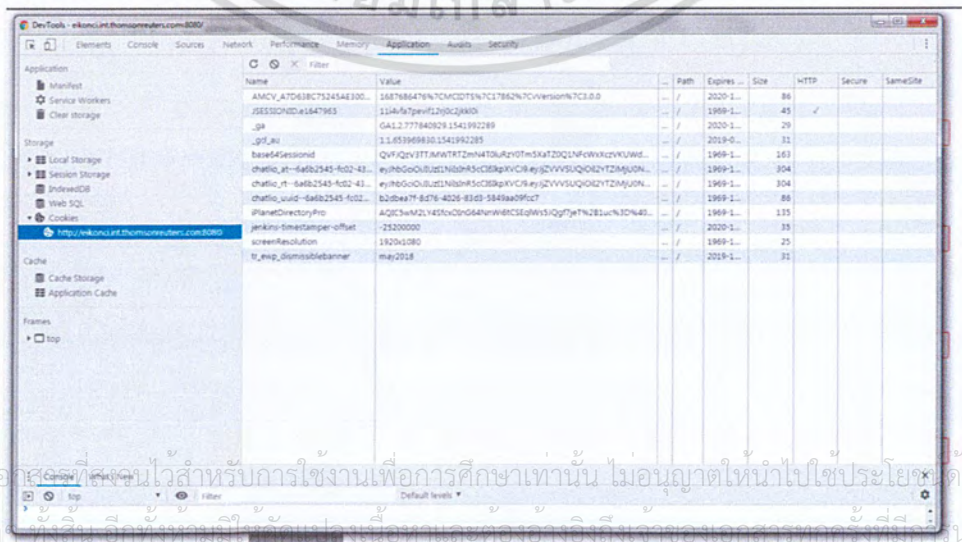


รูป 4.4 เป็นหน้าเว็บแสดงผลการ Login ผิดพลาด หากผู้ใช้กรอก username หรือ password ไม่ถูกต้อง การดำเนินการเข้าสู่ระบบจะไม่สำเร็จ หน้าเว็บจะแสดงข้อความ Username or Password is incorrect และรีเซ็ตฟอร์มให้กรอกใหม่อีกครั้ง



รูปที่ 4.5 หน้าเว็บแสดงผลหน้าหลักหลังจากการ login สำเร็จ

รูป 4.5 เป็นหน้าเว็บแสดงผลหน้าหลักหลังจากการ login สำเร็จ หากผู้ใช้กรอก username และ password ถูกต้อง หลังจากการดำเนินการ login สำเร็จ ระบบจะแสดงหน้าหลักของแอปพลิเคชัน โดยจะแสดงปุ่มให้เลือกตามสิทธิ์ของผู้ใช้ หากผู้ใช้เปิด Chrome DevTools จะพบว่ามีการเก็บ session id ไว้ใน cookie ดังรูป 4.6



รูปที่ 4.6 หน้าเว็บแสดงผลการเก็บ cookie หลังจากการ login สำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือทำซ้ำโดยไม่ได้รับอนุญาตจากเจ้าของลิขสิทธิ์

AppRelease Create Release Logout

Create Release

AppName

AppVersion

ReleaseNote

Submit

App Release Tool of Fofitsoft Refresh Page

รูปที่ 4.7 หน้าเว็บของการ create release

รูป 4.7 เป็นหน้า create release ผู้ใช้ที่เข้าสู่ระบบจะมีปุ่ม Create Release มุมบนด้านขวาของเว็บ เมื่อผู้ใช้เลือกที่ปุ่มดังกล่าว ระบบจะแสดงหน้า create release ให้ผู้ใช้สามารถกรอกชื่อ เวอร์ชัน และ release note ของแอปพลิเคชันที่ต้องการ release เมื่อผู้ใช้ทำการกรอกข้อมูลครบทุกช่อง ปุ่ม submit จะแสดงขึ้นให้สามารถกดปุ่ม submit ได้ ดังรูป 4.8

AppRelease Create Release Logout

Create Release

AppName  
 AdipatWebDemo

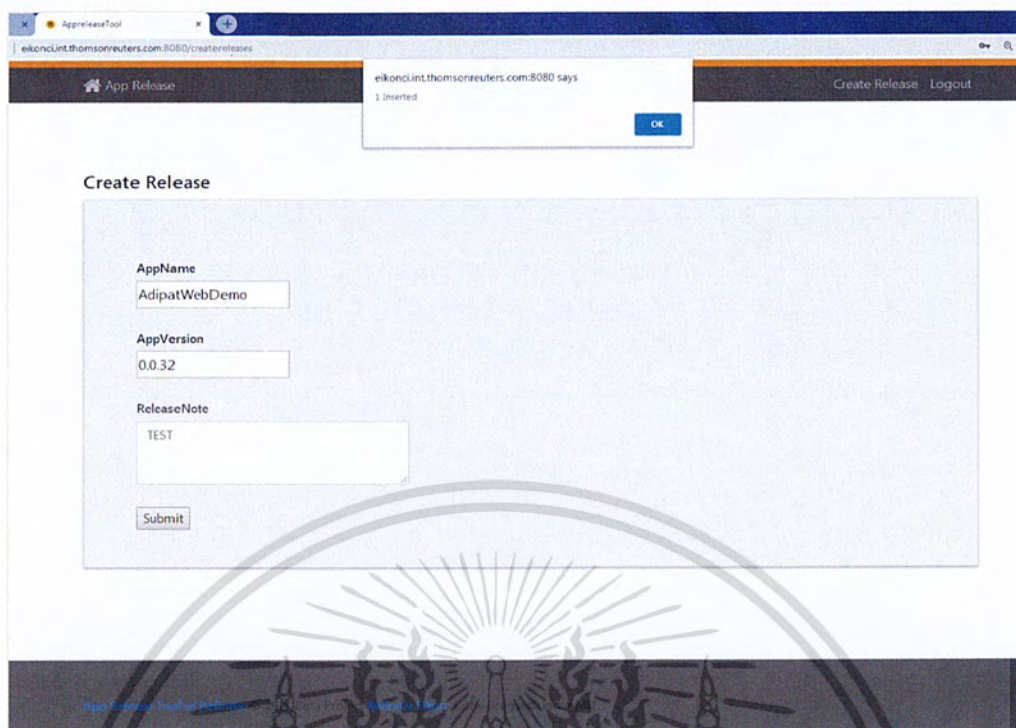
AppVersion  
 0.0.32

ReleaseNote  
 TEST

Submit

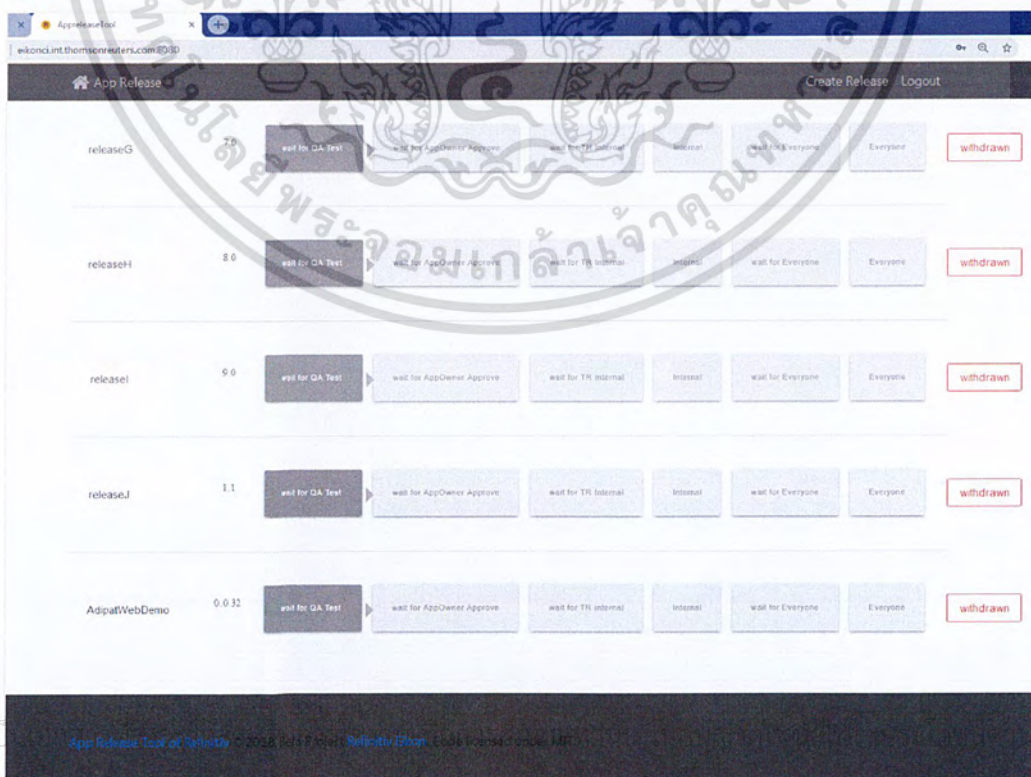
App Release Tool of Fofitsoft Refresh Page

รูปที่ 4.8 หน้าเว็บแสดงการกรอกข้อมูลแอปพลิเคชันที่ต้องการ release

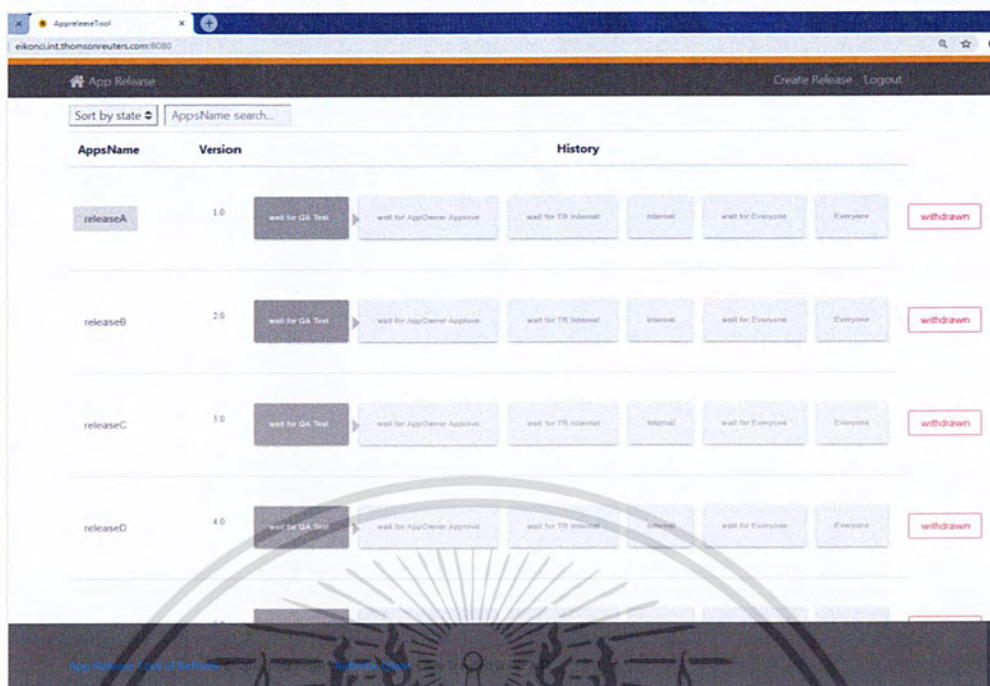


รูปที่ 4.9 หน้าเว็บแสดงผลการ create release สำเร็จ

รูป 4.9 เป็นหน้าเว็บแสดงผลการ create release สำเร็จ เมื่อผู้ใช้ทำการ create release สำเร็จ ระบบจะขึ้นแจ้งว่า 1 inserted ให้ผู้ใช้ทราบ เมื่อผู้ใช้ทำการกดปุ่ม ok ระบบจะแสดงหน้าหลัก โดยจะแสดงรายการ release ทั้งหมดรวมถึงแอปพลิเคชันที่เพิ่งทำการ create release ด้วยดังรูป 4.10

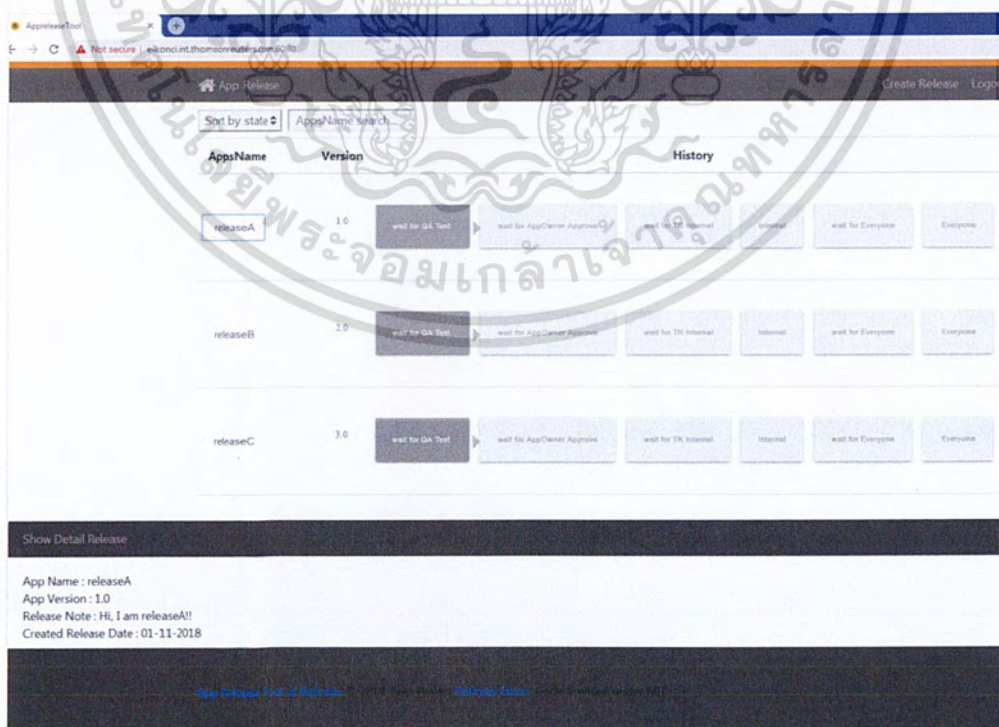


รูปที่ 4.10 หน้าเว็บรายการ release ทั้งหมด หลังการ create release สำเร็จ

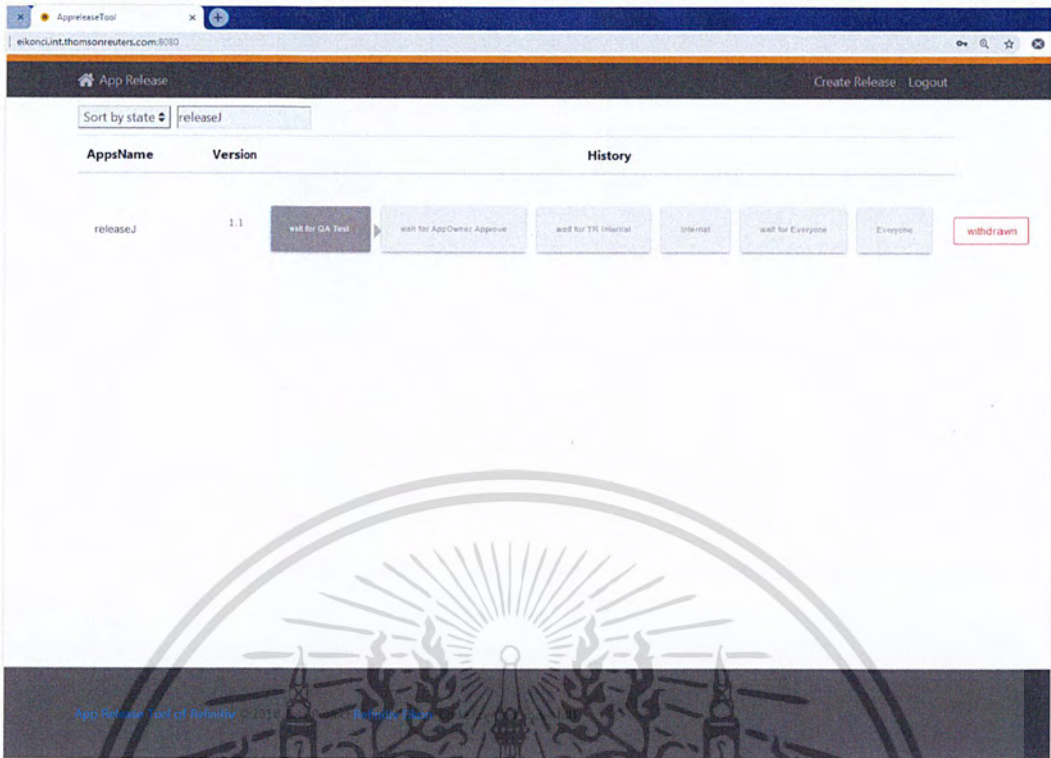


รูปที่ 4.11 หน้าเว็บแสดงการเลือกที่ชื่อของแอปพลิเคชัน

เมื่อผู้ใช้งานต้องการดูรายละเอียดของแต่ละแอปพลิเคชัน สามารถทำได้โดยการเลือกที่ชื่อของแอปพลิเคชันนั้นๆ ดังตัวอย่างรูป 4.11 ทำการเลือกที่ชื่อ releaseA เมื่อผู้ใช้เลือกที่ปุ่มดังกล่าว ระบบจะแสดงรายละเอียดของแอปพลิเคชันนั้น ดังรูป 4.12

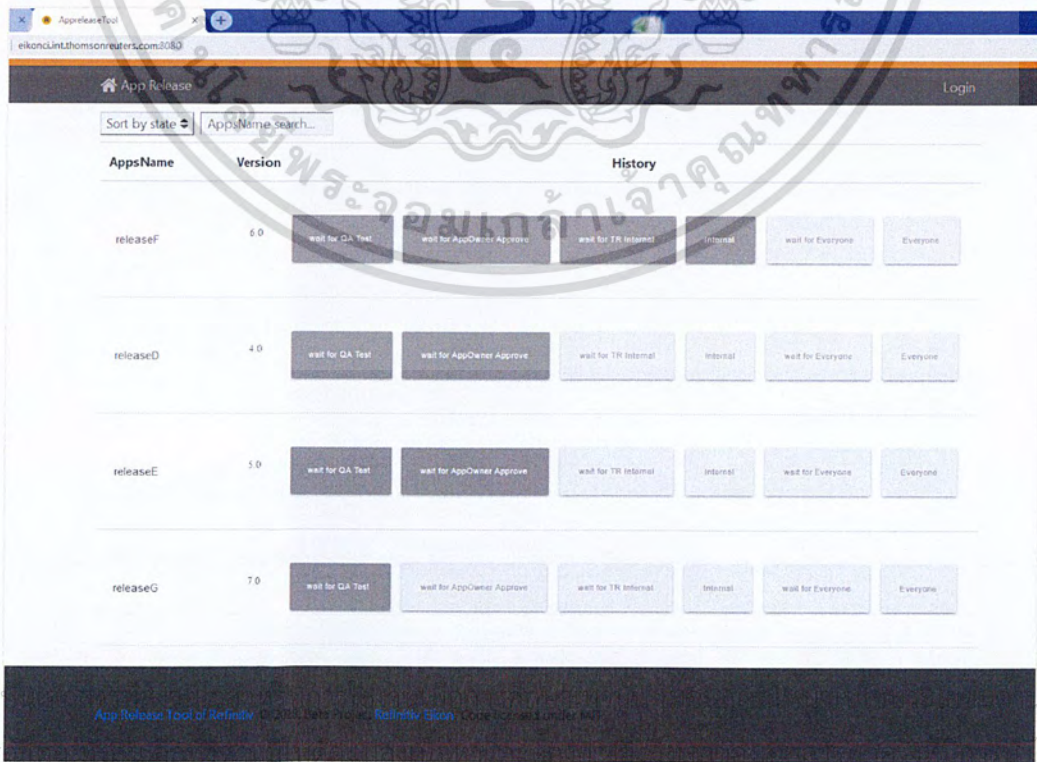


เอกสารนี้เป็นเอกสารทสวทวสวไรสาหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญตให้นำเป็เซประยชนดานการค้  
 ไม่ว่การฉีใด ๆ ทั้งสิ้น รูปที่ 4.12 หน้าเว็บแสดงังรายละเอียดของแอปพลิเคชัน releaseA ั้งที่มีการนำไปใช้



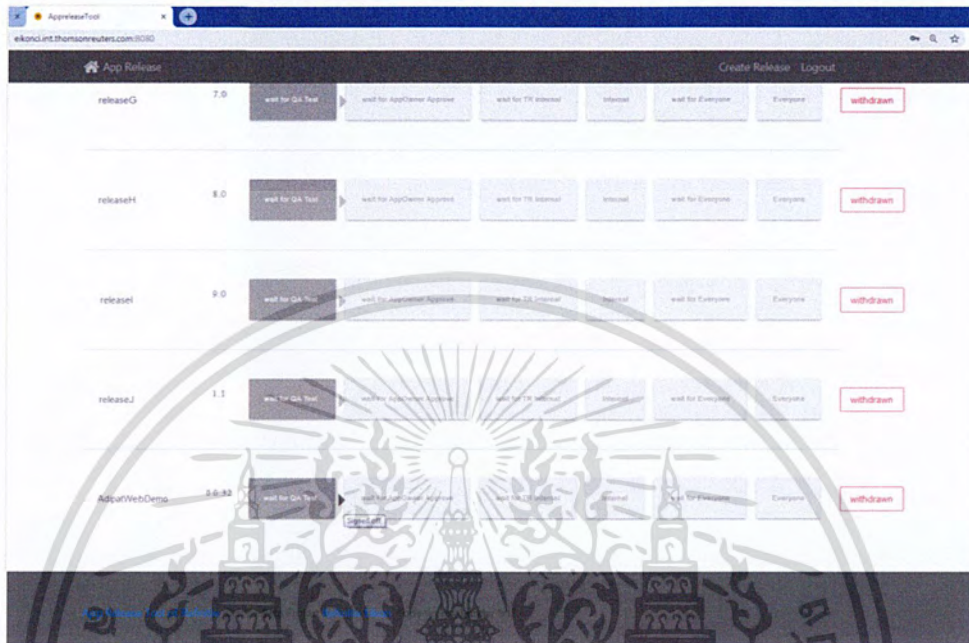
รูปที่ 4.13 หน้าเว็บแสดงการค้นหารายชื่อแอปพลิเคชันที่มีการ create release

รูป 4.13 เป็นหน้าเว็บแสดงการค้นหารายชื่อแอปพลิเคชันที่มีการ create release หลังจากผู้ใช้ทำการกรอกชื่อแอปพลิเคชันที่ช่อง AppsName search บนหน้าหลักของระบบ ระบบจะแสดงรายการของแอปพลิเคชันดังกล่าวให้ผู้ใช้ทราบ



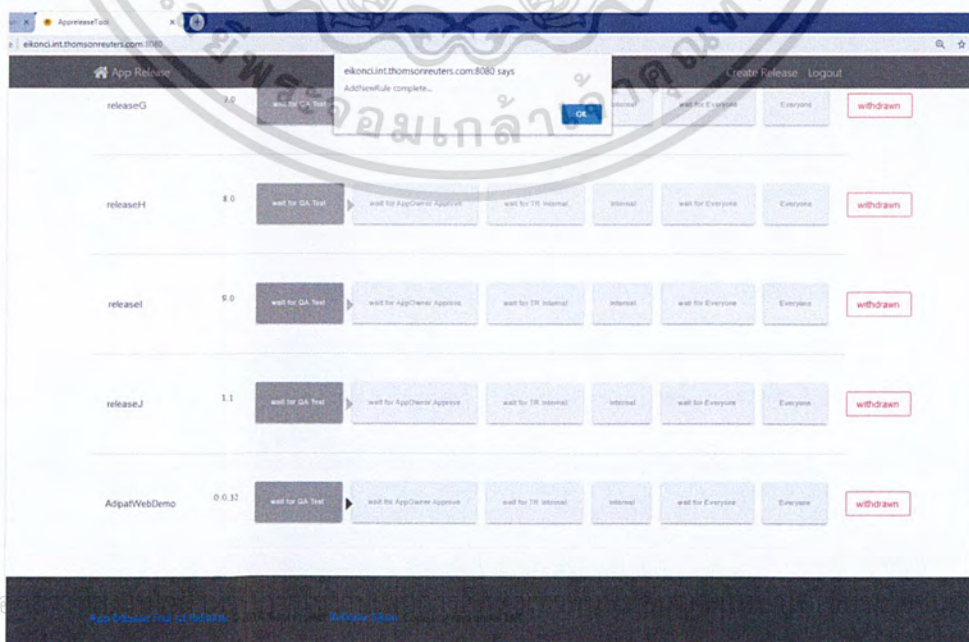
รูปที่ 4.14 หน้าเว็บแสดงรายการ release ทั้งหมด หลังการ sort by state

รูป 4.14 เป็นหน้าเว็บแสดงรายการ release ทั้งหมด หลังการ sort by state หลังจากผู้ใช้ทำการเลือกที่ปุ่ม sort by state บนหน้าหลักของระบบ ระบบจะทำการเรียงลำดับการแสดงผลตามสถานะของแอปพลิเคชันที่มีการ release ให้ผู้ใช้ทราบ



รูปที่ 4.15 หน้าเว็บแสดงปุ่มสำหรับกด signed off แอปพลิเคชันที่มีการ release

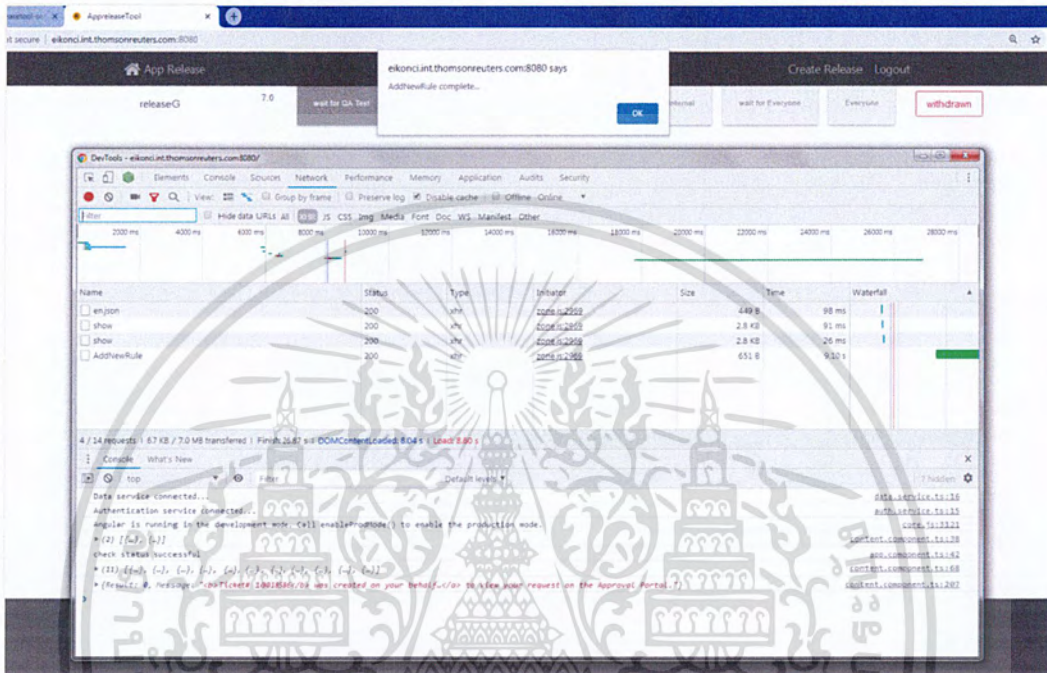
รูป 4.15 เป็นหน้าเว็บแสดงปุ่มสำหรับกด signed off แอปพลิเคชันที่มีการ release ผู้ใช้ที่มีสิทธิ์ signed off หน้าเว็บจะแสดงปุ่มลูกศรให้ผู้ใช้เลือกกด เมื่อผู้ใช้เลื่อนเมาส์ไปวางยังลูกศรดังกล่าว จะมีข้อความ signed off แสดงขึ้นดังรูป



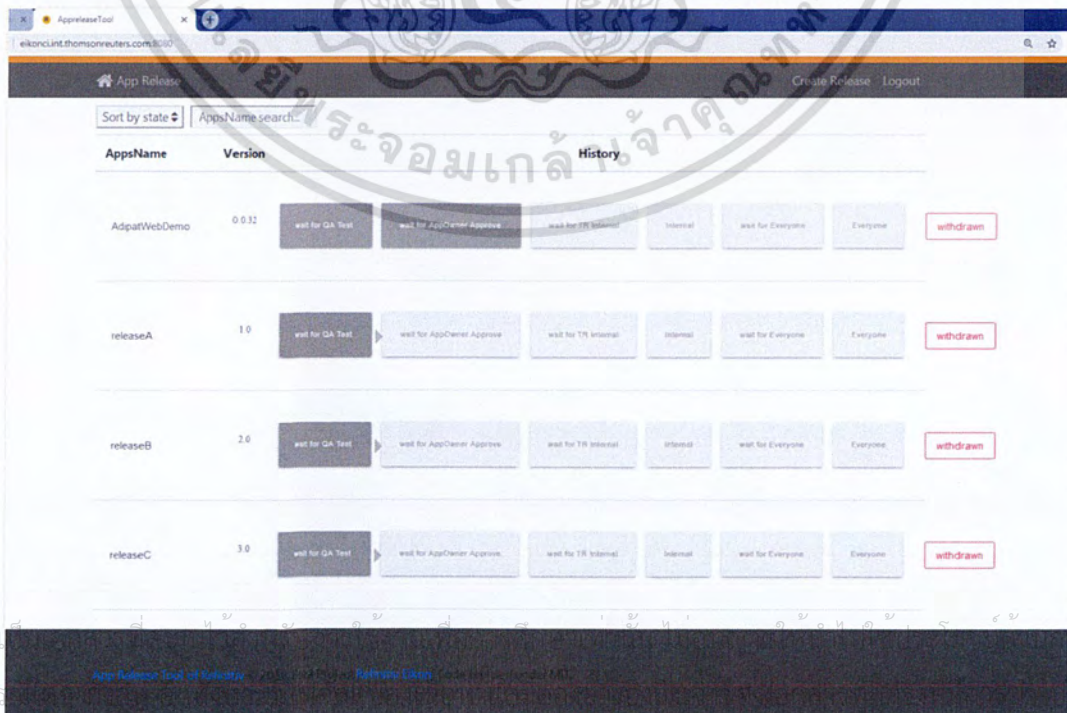
เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการดำเนินงานของหน่วยงานราชการ การนำเอกสารนี้ไปใช้โดยไม่ผ่านการอนุมัติจากทางต้นสังกัดถือว่าผิดระเบียบและต้องรับผิดชอบต่อเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.16 หน้าเว็บแสดงผลหลังจากกดปุ่ม signed off

รูป 4.16 เป็นหน้าเว็บแสดงผลหลังการกดปุ่ม sign off เมื่อผู้ใช้เลือกปุ่ม signed off ระบบจะทำการส่งคำร้องขอ addNewRule ไปยังระบบ AppReleaseJ เมื่อการร้องขอสำเร็จ ระบบจะแสดงข้อความ AddNewRule complete... ดังรูป หากผู้ใช้เปิด Chrome DevTools จะพบว่ามีการส่ง response เลข ticket ID ที่ได้หลังจากการ addNewRule สำเร็จ ดังรูป 4.17

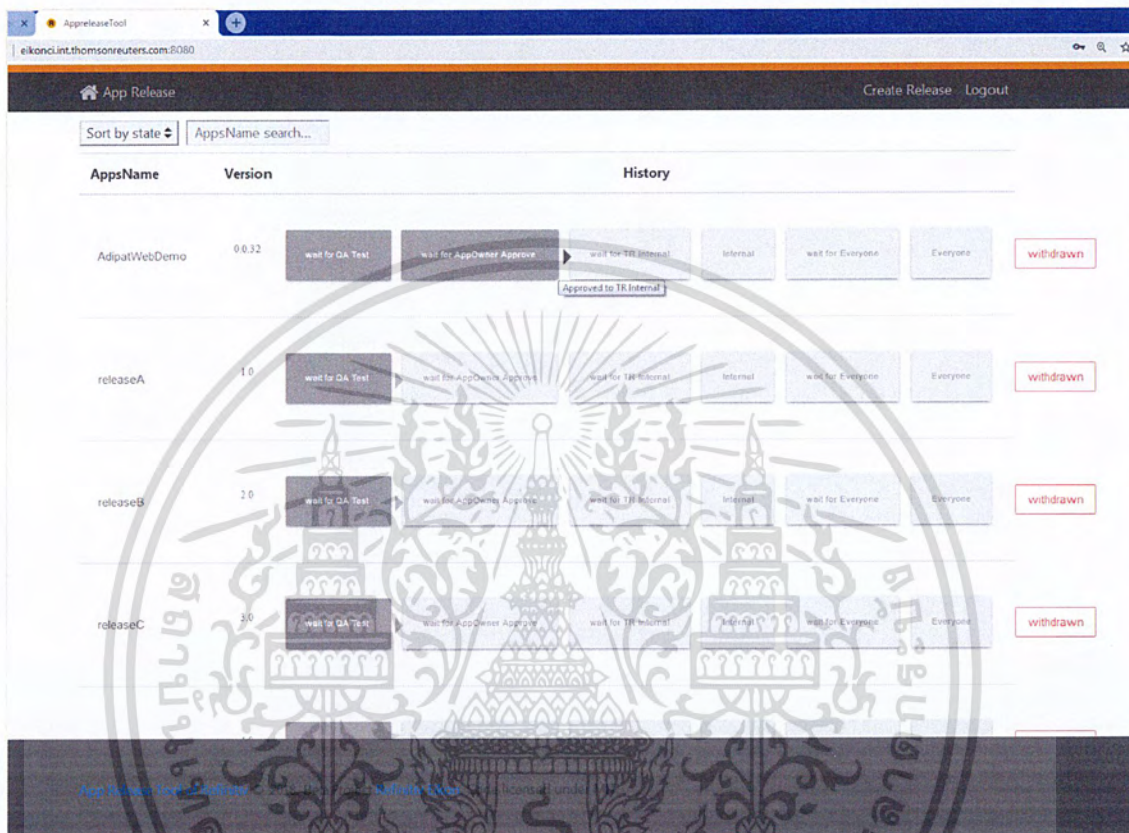


รูปที่ 4.17 หน้าเว็บแสดง response จาก AppReleaseJ หลังจากการ addNewRule สำเร็จ



รูปที่ 4.18 หน้าเว็บแสดงผลหลังการ signed off สำเร็จ

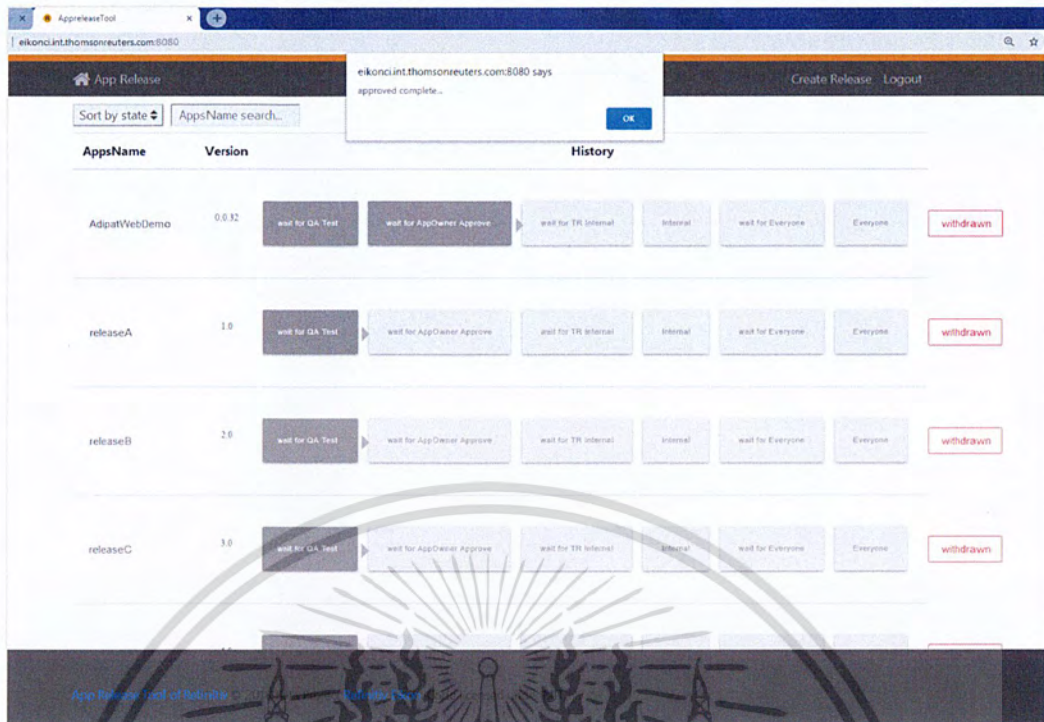
รูป 4.18 เป็นหน้าเว็บแสดงผลการ signed off สำเร็จ เมื่อผู้ใช้ทำการคลิกปุ่ม ok บนหน้าเว็บรูป 4.16 ระบบจะทำการอัปเดตสถานะใหม่จาก wait for QA test เป็น wait for AppOwner Approve ให้ผู้ใช้ได้ทราบ



รูปที่ 4.19 หน้าเว็บแสดงปุ่มสำหรับกด approved แอปพลิเคชันที่มีการ release

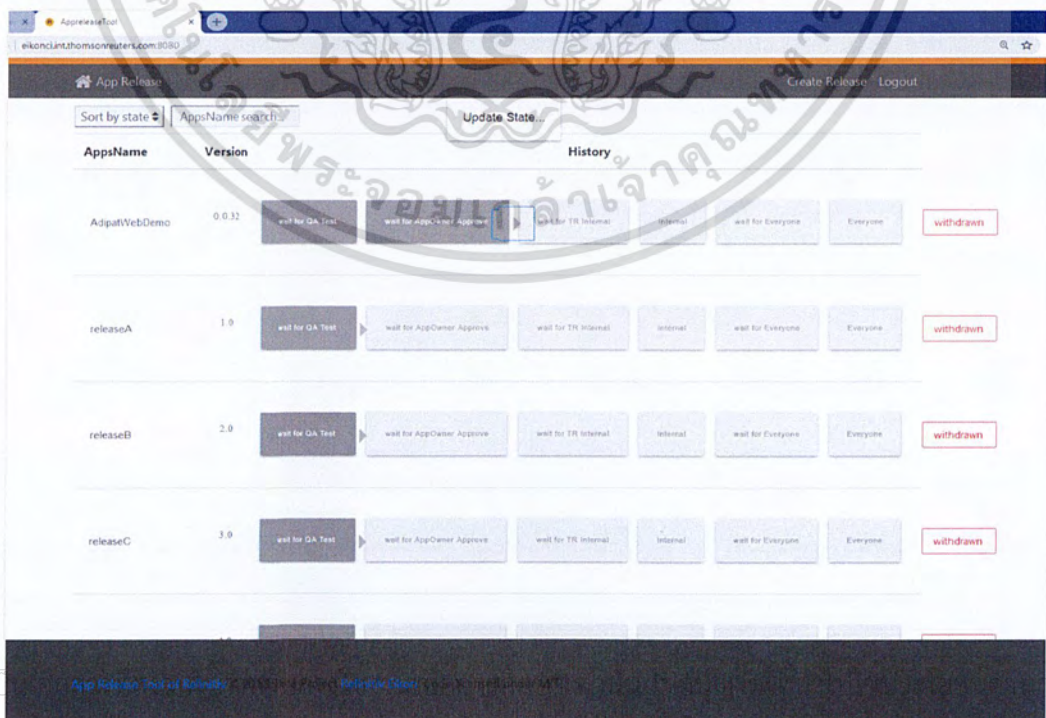
รูป 4.19 เป็นหน้าเว็บแสดงปุ่มสำหรับกด approved แอปพลิเคชันที่มีการ release ผู้ใช้ที่มีสิทธิ์ approved หน้าเว็บจะแสดงปุ่มลูกศรให้ผู้ใช้เลือกกด เมื่อผู้ใช้เลื่อนเมาส์ไปวางยังลูกศรดังกล่าว จะมีข้อความ Approved to TR Internal แสดงขึ้นดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

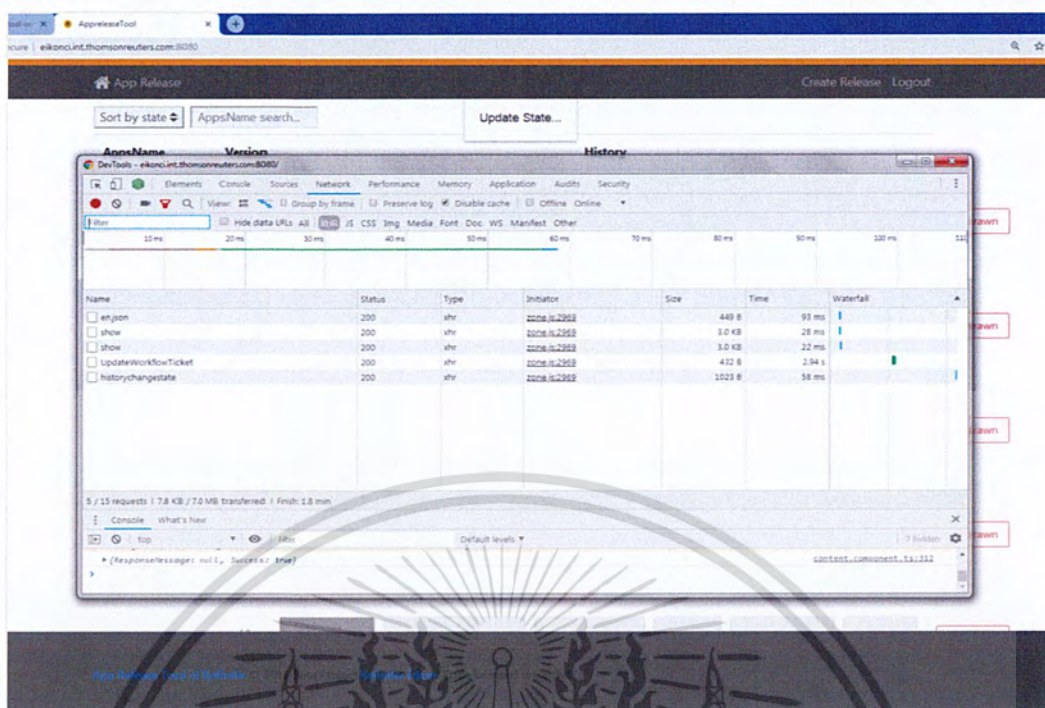


รูปที่ 4.20 หน้าเว็บแสดงผลหลังจากกดปุ่ม approved

รูป 4.20 เป็นหน้าเว็บแสดงผลหลังการกดปุ่ม approved เมื่อผู้ใช้เลือกปุ่ม approved ระบบจะทำการส่งคำร้องขอ updateWorkflowTicket ไปยังระบบ AppReleaseJ เมื่อการร้องขอสำเร็จ ระบบจะแสดงข้อความ approved complete... ดังรูป เมื่อผู้ใช้ทำการคลิกปุ่ม ok ระบบจะทำการอัปเดตสถานะใหม่ โดยจะขึ้นข้อความ Update State... ให้ผู้ใช้ทราบดังรูป 4.21



รูปที่ 4.21 หน้าเว็บแสดงข้อความ Update State หลังจากการคลิกปุ่ม ok

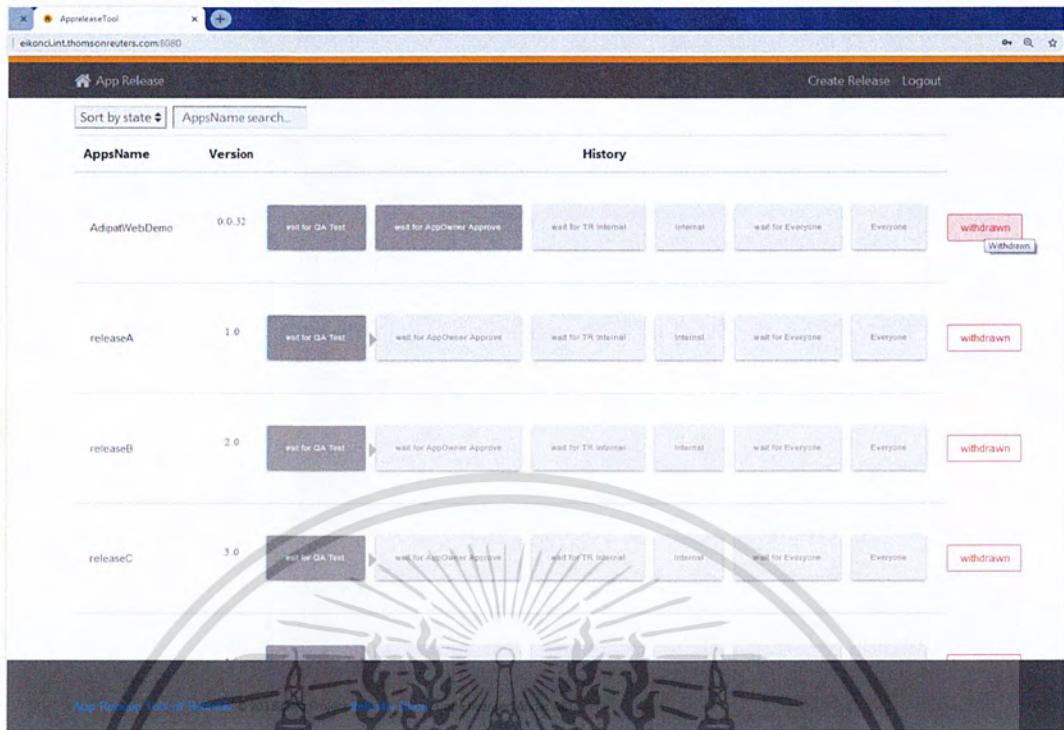


รูปที่ 4.22 หน้าเว็บแสดงการร้องขอ UpdateWorkflowTicket และ HistoryChangeState

รูป 4.22 เป็นหน้าเว็บแสดงการส่ง request เพื่ออัปเดต WorkflowTicket และการส่ง request เพื่อเปลี่ยนสถานะ เมื่อผู้ใช้เปิด Chrome DevTools ถ้าการร้องขอสำเร็จ ระบบจะได้รับ response กลับเป็น Status Code 200 และเมื่อการอัปเดตสถานะสำเร็จระบบจะแสดงสถานะใหม่จาก wait for AppOwner Approve เป็น wait for TR Internal ให้ผู้ใช้ได้ทราบดังรูป 4.23

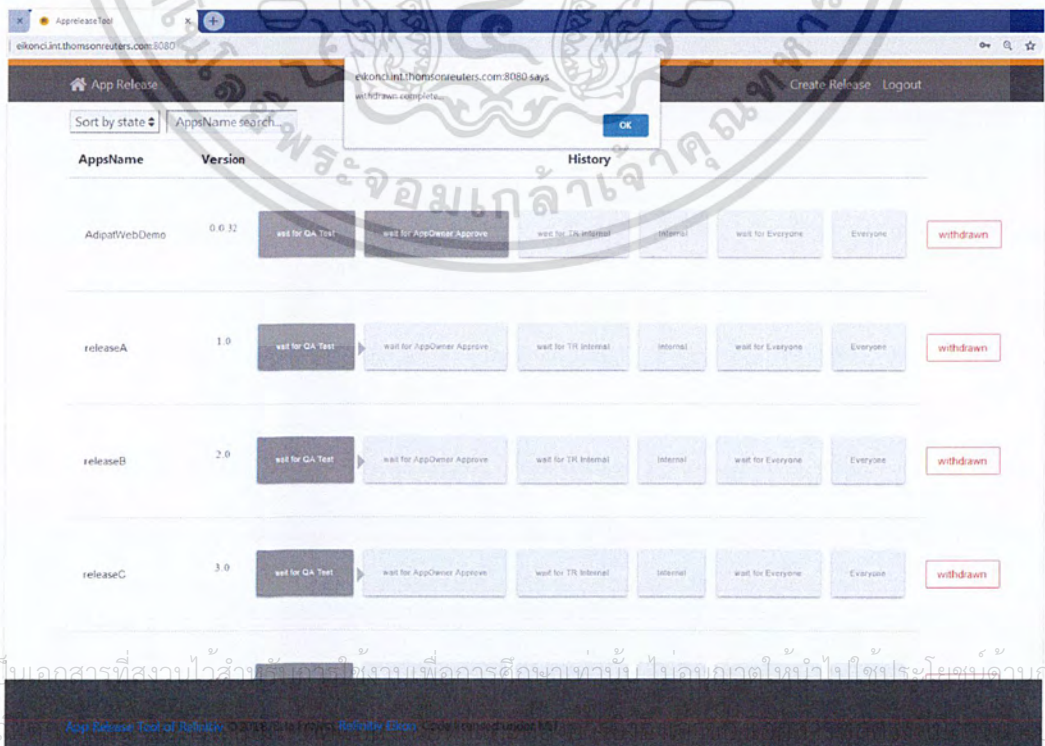


รูปที่ 4.23 หน้าเว็บแสดงผลหลังการ approved สำเร็จ

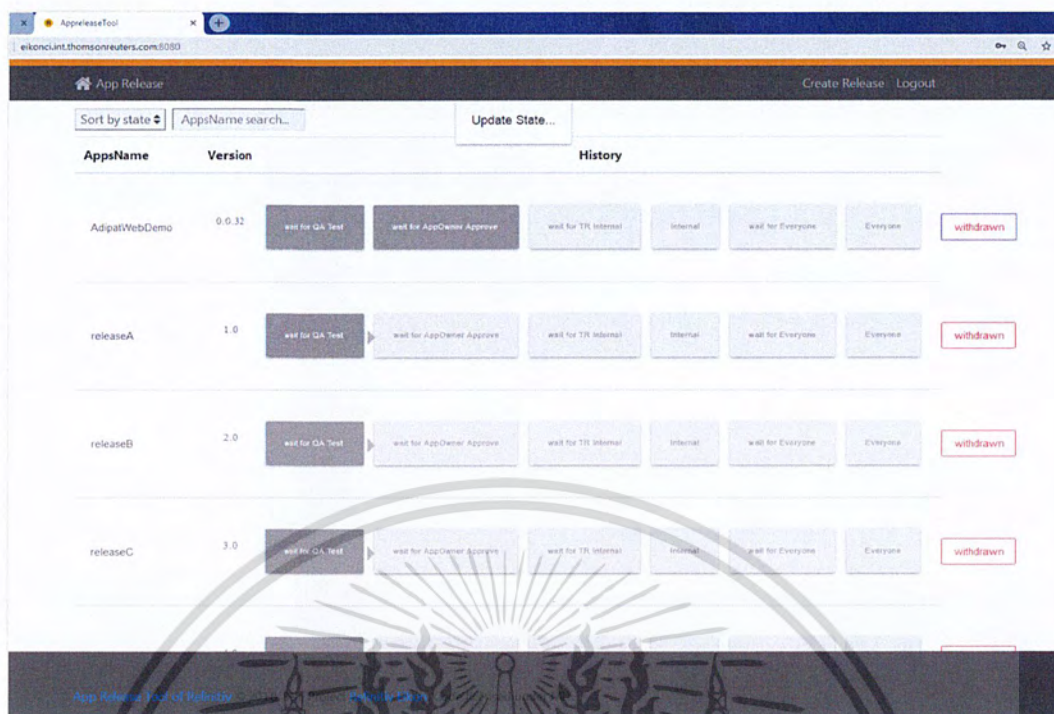


รูปที่ 4.24 หน้าเว็บแสดงปุ่มสำหรับกด withdrawn แอปพลิเคชันที่มีการ release

รูป 4.24 เป็นหน้าเว็บแสดงปุ่มสำหรับกด withdraw แอปพลิเคชันที่มีการ release ผู้ใช้ที่มีสิทธิ์ withdraw หน้าเว็บจะแสดงปุ่มผู้ใช้เลือกกด เมื่อผู้ใช้เลื่อนเมาส์ไปวางยังปุ่มดังกล่าว จะมีข้อความ Withdrawn แสดงขึ้นดังรูป และหลังจากผู้ใช้กดปุ่ม withdraw ระบบจะแสดงข้อความ withdrawn complete... ดังรูป 4.25

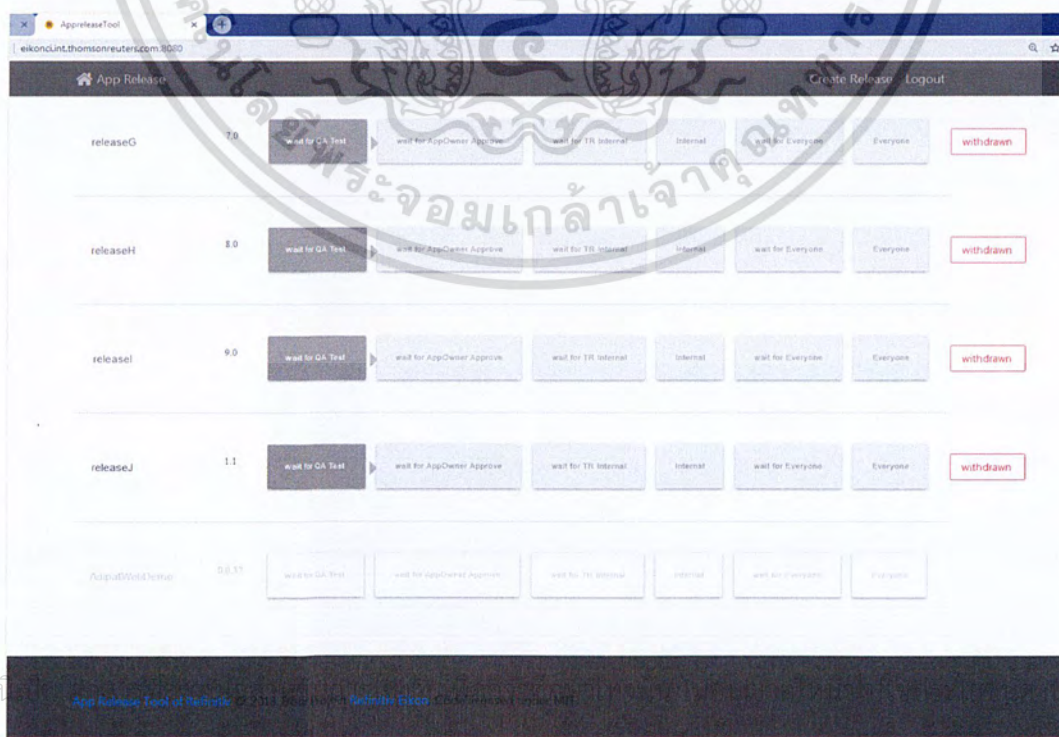


รูปที่ 4.25 หน้าเว็บแสดงผลหลังจากกดปุ่ม withdrawn

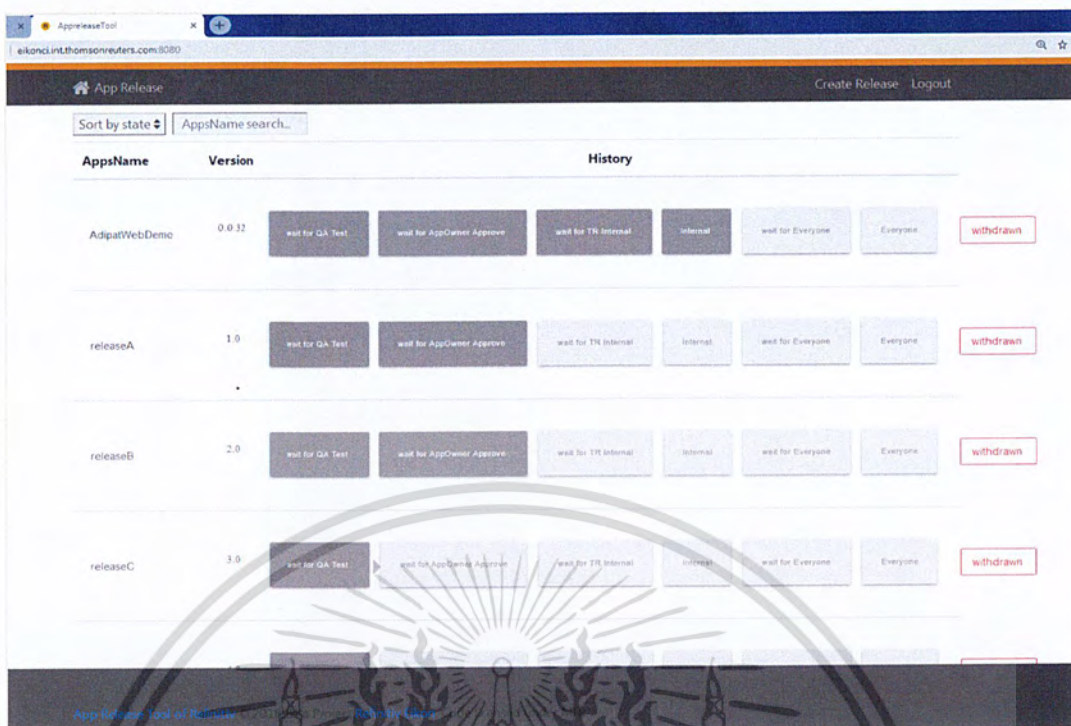


รูปที่ 4.26 หน้าเว็บแสดงข้อความ Update State หลังจากการคลิกปุ่ม ok

รูป 4.26 เป็นหน้าเว็บแสดงข้อความ Update State หลังจากการคลิกปุ่ม ok เมื่อผู้ใช้ทำการคลิกปุ่ม ok ในรูป 4.25 ระบบจะทำการอัปเดตสถานะใหม่ โดยจะขึ้นข้อความ Update State... ให้ผู้ใช้ทราบ และเมื่อการอัปเดตสถานะสำเร็จระบบจะแสดงสถานะใหม่ โดยแอปพลิเคชันที่ถูก withdrawn สำเร็จ จะถูกเปลี่ยนสีเป็นสีขาวเทาเพื่อให้สามารถแยกออกได้ง่ายขึ้น ดังรูป 4.27

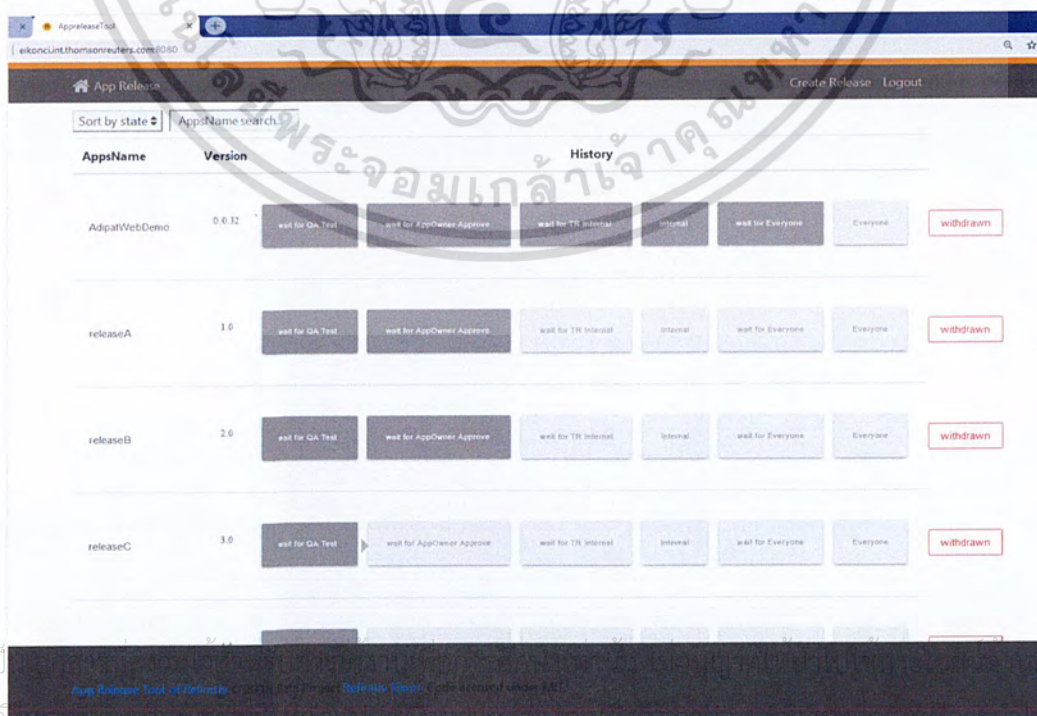


รูปที่ 4.27 หน้าเว็บแสดงผลหลังการ withdrawn สำเร็จ



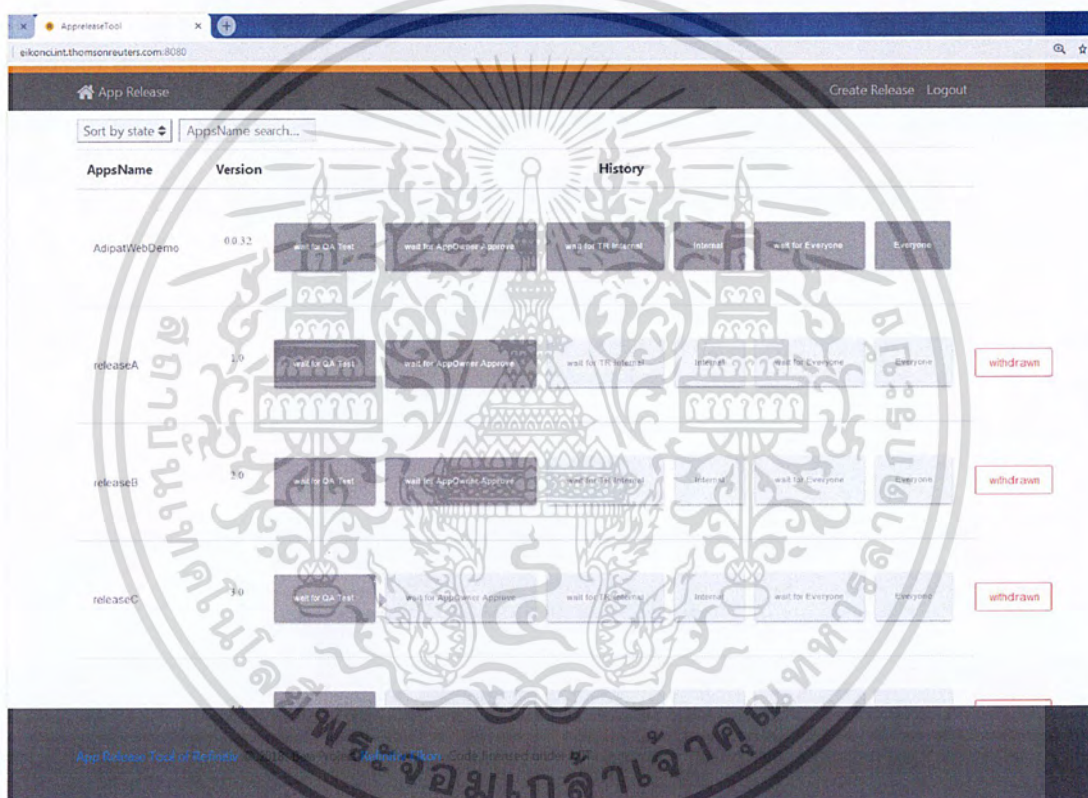
รูปที่ 4.28 หน้าเว็บแสดงสถานะ Internal ของแอปพลิเคชัน AdipatWebDemo

รูป 4.28 เป็นการแสดงสถานะ Internal ของแอปพลิเคชัน AdipatWebDemo เมื่อระบบทำการตรวจสอบสถานะจริงในระบบ AppRelease แล้วพบว่าสถานะจริงถูก release ไปยัง Internal เรียบร้อยแล้ว ระบบเว็บจะดำเนินการอัปเดตสถานะใหม่ จาก wait for TR Internal เป็น Internal ซึ่งการดำเนินการจะทำได้โดยอัตโนมัติ และถูกแสดงบนหน้าเว็บให้ผู้ใช้ทราบดังรูป



รูปที่ 4.29 หน้าเว็บแสดงสถานะ wait for Everyone ของแอปพลิเคชัน AdipatWebDemo

รูป 4.29 เป็นการแสดงสถานะ wait for Everyone ของแอปพลิเคชัน AdipatWebDemo เมื่อระบบทำการตรวจสอบสถานะจริงในระบบ AppReleaseJ แล้วพบว่าสถานะจริงถูก release ไปยัง Internal เรียบร้อยแล้ว ระบบเว็บจะดำเนินการร้องขอ UpdateWorkFlowTicket ไปยังระบบ AppReleaseJ เป็นการ approved release ไปยัง everyone โดยอัตโนมัติ เมื่อระบบทำการตรวจสอบสถานะจริงในระบบ AppReleaseJ แล้วพบว่าสถานะจริงกำลังถูกดำเนินการ release ไป everyone ระบบจะทำการอัปเดตสถานะใหม่ จาก Internal เป็น wait for Everyone ซึ่งการดำเนินการจะทำโดยอัตโนมัติ และถูกแสดงบนหน้าเว็บให้ผู้ใช้ทราบดังรูป



รูปที่ 4.30 หน้าเว็บแสดงสถานะ Everyone ของแอปพลิเคชัน AdipatWebDemo

รูป 4.30 เป็นการแสดงสถานะ Everyone ของแอปพลิเคชัน AdipatWebDemo เมื่อระบบทำการตรวจสอบสถานะจริงในระบบ AppReleaseJ แล้วพบว่าสถานะจริงถูก release ไปยัง Everyone เรียบร้อยแล้ว ระบบเว็บจะดำเนินการอัปเดตสถานะใหม่ จาก wait for Everyone เป็น Everyone ซึ่งการดำเนินการจะทำโดยอัตโนมัติ และถูกแสดงบนหน้าเว็บให้ผู้ใช้ทราบดังรูป ( หากแอปพลิเคชัน ถูก release ไปยัง everyone แล้ว แอปพลิเคชันดังกล่าวจะไม่สามารถ withdrawn ได้ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 ผลลัพธ์การทดสอบระบบแอปพลิเคชันของระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้น ภายในองค์กรของบริษัท Refinitiv

หลังจากการพัฒนาระบบสำเร็จ จำเป็นต้องมีการทดสอบระบบเว็บแอปพลิเคชัน เพื่อทดสอบว่าระบบสามารถทำงานและแสดงผลลัพธ์ได้ถูกต้องตามที่ต้องการหรือไม่ ในการทดสอบระบบจะทำการทดสอบโดยใช้วิธีการทดสอบแบบ Functional testing (Black box testing) มีการกำหนดกรณีทดสอบ (Test Case) ซึ่งจะเน้นที่การทดสอบจาก User Interface ของระบบ

การทดสอบระบบดำเนินการตาม Feature ดังต่อไปนี้

- 1) ลงชื่อเข้าใช้งานระบบ ( Login )
- 2) สร้างรายชื่อแอปพลิเคชันที่ต้องการร้องขอ release ( Create release )
- 3) Signed off แอปพลิเคชันที่มีการร้องขอ release ( Signed off release )
- 4) อนุมัติแอปพลิเคชันที่ต้องการ release ไป internal ( Approved release )
- 5) ยกเลิกการ release แอปพลิเคชัน ( Withdrawn release )
- 6) ดูรายละเอียดแอปพลิเคชันที่มีการ release ไป everyone ( View detail release )
- 7) ค้นหารายชื่อแอปพลิเคชันที่มีการ create release ( Search release by application name )
- 8) เรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชันที่มีการ release ( Sort by state )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1) ลงชื่อเข้าใช้งานระบบ (Login)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 3 กรณีดังนี้

1. กรอก username และ password ถูกต้อง
2. กรอก username หรือ password ไม่ถูกต้อง
3. ไม่กรอก username และ password

ตารางที่ 4.1 ผลการทดสอบฟังก์ชันลงชื่อเข้าใช้งานระบบ

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กรอก username และ password ถูกต้อง	Username /password ที่ใช้ login Eikon	1.กรอก username	Siriporn.Kiwatthana2@thomsonreuters.com	login สำเร็จ navigate ไปยังหน้าหลัก	Pass
		2.กรอก password	newnewnew1		
		3.กดปุ่ม Login			
กรอก username หรือ password ไม่ถูกต้อง		1.กรอก username	aaaaa	แสดงข้อความแจ้ง "username or password is incorrect"	Pass
		2.กรอก password	aaaaa		
		3.กดปุ่ม Login			
ไม่กรอก username และ password		1.ไม่กรอก username		แสดงข้อความแจ้ง "username or password is incorrect"	Pass
		2.ไม่กรอก password			
		3.กดปุ่ม Login			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2) สร้างรายชื่อแอปพลิเคชันที่ต้องการร้องขอ release (Create release)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 3 กรณีดังนี้

1. กรอก AppName AppVersion ReleaseNote ครบทุกช่อง
2. กรอก AppName AppVersion ReleaseNote ไม่ครบทุกช่อง
3. ไม่กรอก AppName, AppVersion และ ReleaseNote

ตารางที่ 4.2 ผลการทดสอบฟังก์ชันสร้างรายชื่อแอปพลิเคชันที่ต้องการร้องขอ release

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กรอก AppName AppVersion ReleaseNote ครบทุกช่อง	AppName ,AppVersion, ReleaseNote ที่ต้องการร้องขอ Release	1.กรอก AppName 2.กรอก AppVersion 3.กรอก ReleaseNote 4.กดปุ่ม submit	AdipatWebDemo 0.0.32 test	-ขึ้น alert box สร้างรายชื่อสำเร็จ - click ปุ่ม ok ไปยังหน้าหลัก	Pass
กรอก AppName AppVersion ReleaseNote ไม่ครบทุกช่อง		1.ไม่กรอก AppName 2.กรอก AppVersion 3.กรอก ReleaseNote 4.กดปุ่ม submit	0.0.32 test	ไม่สามารถกดปุ่ม submit ได้ เนื่องจากแสดงปุ่มเป็น ประเภท Disabled Button	Pass
ไม่กรอก AppName, AppVersion และ ReleaseNote		1.ไม่กรอก AppName 2.ไม่กรอก AppVersion 3.ไม่กรอก ReleaseNote 4.กดปุ่ม submit		ไม่สามารถกดปุ่ม submit ได้ เนื่องจากแสดงปุ่มเป็น ประเภท Disabled Button	Pass

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3) Signed off แอปพลิเคชันที่มีการร้องขอ release (Signed off release)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 2 กรณีดังนี้

1. กดปุ่ม Signed off
2. ไม่กดปุ่ม Signed off

## ตารางที่ 4.3 ผลการทดสอบฟังก์ชัน Signed off แอปพลิเคชันที่มีการร้องขอ release

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กดปุ่ม Signed off	ปุ่ม Signed off จะแสดงให้เห็นสำหรับ User ที่มีสิทธิ์ Signed off เท่านั้น	กดปุ่ม Signed off		-ขึ้น alert box “AddNewRule complete...” - click ปุ่ม ok ระบบทำการอัปเดตสถานะเป็น “wait for AppOwner Approve”	Pass
ไม่กดปุ่ม Signed off		ไม่กดปุ่ม Signed off		การ Signed off release ไม่สำเร็จ	Pass

## 4) อนุมัติแอปพลิเคชันที่ต้องการ release ไป internal (Approved release)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 2 กรณีดังนี้

1. กดปุ่ม Approved
2. ไม่กดปุ่ม Approved

## ตารางที่ 4.4 ผลการทดสอบฟังก์ชันอนุมัติแอปพลิเคชันที่ต้องการ release ไป internal

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กดปุ่ม Approved	ปุ่ม Approved จะแสดงให้เห็นสำหรับ User ที่มีสิทธิ์ Approved เท่านั้น	กดปุ่ม Approved		-ขึ้น alert box “approved complete...” - click ปุ่ม ok ระบบทำการอัปเดตสถานะเป็น “wait for TR Internal”	Pass
ไม่กดปุ่ม Approved		ไม่กดปุ่ม Approved		การ Approved release ไม่สำเร็จ	Pass

## 5) ยกเลิกการ release แอปพลิเคชัน (Withdrawn release)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 2 กรณีดังนี้

1. กดปุ่ม Withdrawn
2. ไม่กดปุ่ม Withdrawn

## ตารางที่ 4.5 ผลการทดสอบฟังก์ชันยกเลิกการ release แอปพลิเคชัน

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กดปุ่ม Withdrawn	ปุ่ม Withdrawn จะแสดงให้เห็นสำหรับ User ที่มีสิทธิ์ Withdrawn เท่านั้น	กดปุ่ม Withdrawn		-ขึ้น alert box “withdrawn complete...” - click ปุ่ม ok ระบบทำการอัปเดตสถานะ แอปพลิเคชันที่ถูก withdrawn สำเร็จ จะถูกเปลี่ยนสีเป็นสีขาวเทา	Pass
ไม่กดปุ่ม Withdrawn		ไม่กดปุ่ม Withdrawn		การ Withdrawn release ไม่สำเร็จ	Pass

## 6) ดูรายละเอียดแอปพลิเคชันที่มีการร้องขอ release ไป everyone (View detail release)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 2 กรณีดังนี้

1. กดปุ่ม AppName ที่ต้องการดูรายละเอียด
2. ไม่กดปุ่ม AppName ที่ต้องการดูรายละเอียด

## ตารางที่ 4.6 ผลการทดสอบฟังก์ชัน View detail release

Test Cases	Description	Test Step	Expected Result	Actual Result
กดปุ่ม AppName ที่ต้องการดูรายละเอียด	ปุ่ม AppName จะแสดงให้เห็นสำหรับเมื่อใช้เมาส์ไปวางบนชื่อแอปพลิเคชัน	1.เข้าสู่หน้าหลัก	-แสดงรายละเอียดของแอปพลิเคชันที่มีการ release	Pass
		2.clickบนรายชื่อที่ต้องการดูรายละเอียด		
ไม่กดปุ่ม AppName ที่ต้องการดูรายละเอียด		1.เข้าสู่หน้าหลัก	-หน้าต่างแสดง	Pass
		2.ไม่ click บนรายชื่อแอปพลิเคชัน	รายละเอียดของแอปพลิเคชันไม่ปรากฏ	

## 7) ค้นหารายชื่อแอปพลิเคชันที่มีการ create release (Search release by application name)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 3 กรณีดังนี้

1. กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหาที่มีในระบบ
2. กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหาที่ไม่มีในระบบ
3. ไม่กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหา

ตารางที่ 4.7 ผลการทดสอบฟังก์ชันค้นหารายชื่อแอปพลิเคชันที่มีการ create release

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหาที่มีในระบบ	ช่องค้นหาจะแสดงให้เห็นเมื่อเข้าสู่หน้าหลักของระบบ	1.เข้าสู่หน้าหลัก		-หน้าหลักแสดงรายการของแอปพลิเคชันที่ทำการค้นหา	Pass
		2.กรอก AppName บนช่องค้นหา	AdipatWeb Demo		
กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหาที่ไม่มีในระบบ		1.เข้าสู่หน้าหลัก		-หน้าหลักไม่แสดงรายการของแอปพลิเคชันที่ทำการค้นหา	Pass
		2.กรอก AppName บนช่องค้นหา	aaaaaa		
ไม่กรอกรายชื่อแอปพลิเคชันที่ต้องการค้นหา		1.เข้าสู่หน้าหลัก		-หน้าหลักแสดงรายการของแอปพลิเคชันทั้งหมดในระบบ	Pass
		2.ไม่กรอก AppName บนช่องค้นหา			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) เรียงลำดับการแสดงผลหน้าหลักตามสถานะของแอปพลิเคชันที่มีการ release (Sort by state)

ในการทดสอบแบ่งกรณีทดสอบ (Test Cases) ได้เป็น 2 กรณีดังนี้

1. กดปุ่ม Sort by state
2. ไม่กดปุ่ม Sort by state

ตารางที่ 4.8 ผลการทดสอบฟังก์ชัน Sort by state

Test Cases	Description	Test Step	Test Data	Expected Result	Actual Result
กดปุ่ม Sort by state	ปุ่ม Sort by state จะแสดงให้เห็นเมื่อเข้าสู่หน้าหลักของระบบ	1.เข้าสู่หน้าหลัก		-แสดงผลหน้าหลักตามสถานะของแอปพลิเคชันที่มีการ release	Pass
		2.click กดปุ่ม Sort by state		-ถ้าปัจจุบัน state เรียงจากมากไปน้อย ระบบจากการเรียงจากน้อยไปมาก -ถ้าปัจจุบัน state เรียงจากน้อยไปมาก ระบบจากการเรียงจากมากไปน้อย	
ไม่กดปุ่ม Sort by state		1.เข้าสู่หน้าหลัก		-แสดงผลหน้าหลักตามสถานะของแอปพลิเคชันที่มีการ release	Pass
		2.ไม่กดปุ่ม Sort by state		จากมากไปน้อย	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการดำเนินงานในการสร้างระบบ Software Releasing System Inside the Refinitiv Corporation ได้มีการให้ผู้ใช้งานประเมินผล โดยผู้ประเมินผลคือ นายอดิภัทร ลากรัตนากุล ตำแหน่ง Development Manager ดังตารางด้านล่างนี้ โดยในช่องคะแนน 5 = พึงพอใจมากที่สุด, 4 = พึงพอใจมาก, 3 = ปานกลาง, 2 = น้อย และ 1 = น้อยที่สุดหรือไม่พอใจ

ตารางที่ 4.9 การประเมินผลการสร้างระบบ Software Releasing System Inside the Refinitiv Corporation

ลำดับ	รายการการประเมิน	คะแนน
1	วิธีการป้อนข้อมูลนำเข้า	5
2	รูปแบบการใช้งานเข้าใจง่าย	4
3	ความสะดวก เอื้ออำนวยต่อการใช้งาน	4
4	การป้องกันการป้อนข้อมูลผิดพลาด	3
5	ความสวยงามของโปรแกรม GUI	3
6	การจัดการกับข้อผิดพลาดที่เกิดขึ้นได้	4
7	ลักษณะของผลลัพธ์ตรงกับความต้องการของผู้ใช้งาน	5
8	ผลลัพธ์อยู่ในรูปแบบที่สามารถอ่านได้ง่ายและเข้าใจง่าย	5
9	ผลลัพธ์มีความถูกต้องแม่นยำ	5
	เฉลี่ย	4.22

จากตารางที่ 4.9 ทำให้ทราบว่าระบบได้รับผลการประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 4.22 ซึ่งในที่นี้ รายการการประเมินลำดับที่ 4 การป้องกันการป้อนข้อมูลผิดพลาด และรายการการประเมินลำดับที่ 5 ความสวยงามของโปรแกรม GUI ได้รับคะแนนในระดับปานกลาง ซึ่งเป็นเหตุให้การพัฒนาระบบในอนาคตจำเป็นต้องเพิ่มเติมในส่วนดังกล่าวมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลการดำเนินงาน และข้อเสนอแนะ

การทำสหกิจศึกษาในหัวข้อเรื่อง ระบบการปล่อยซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ( Software Releasing System Inside the Refinitiv Corporation ) สามารถสรุปผลการดำเนินโครงการและข้อเสนอแนะ ได้ดังนี้

#### 5.1 สรุปผลการดำเนินงาน

จากการประเมินระบบที่สร้างขึ้น ผลประเมินจากผู้ใช้งานได้ค่าเฉลี่ย 4.22 ระบบสามารถทำงานในกระบวนการร้องขอและการอนุมัติซอฟต์แวร์ที่สร้างขึ้นภายในองค์กรของบริษัท Refinitiv ได้ ดังนี้

- 1) สามารถทำการร้องขอการปล่อยซอฟต์แวร์ที่พัฒนาขึ้นได้ ( การร้องขอ release ) กระบวนการในการร้องขอสามารถดำเนินการได้โดยกดเพียงปุ่มเดียว ทำให้ผู้ร้องขอสะดวกมากยิ่งขึ้น
- 2) ผู้อนุมัติสามารถทำการอนุมัติซอฟต์แวร์ที่มีการร้องขอได้โดยการกดเพียงปุ่มเดียว ทำให้สามารถลดขั้นตอนการอนุมัติ ช่วยให้การอนุมัติสะดวกมากยิ่งขึ้น และเมื่อซอฟต์แวร์ดังกล่าวถูกปล่อยไปให้ผู้ใช้งานภายในองค์กรใช้เรียบร้อยแล้ว ระบบสามารถทำการอนุมัติการร้องขอไปให้ผู้ใช้งานทั่วไปที่ใช้บริการซอฟต์แวร์ Eikon ใช้ได้โดยอัตโนมัติ ทำให้ลดระยะเวลาในการทำงานขั้นตอนการร้องขออนุมัติแบบ manual
- 3) ทำให้ผู้ใช้งานสามารถมองเห็นสถานะทั้งหมดของซอฟต์แวร์ที่กำลังมีการ release ได้อย่างง่ายขึ้น มีการแสดงสถานะการปล่อยแบบเป็นลำดับขั้น

#### 5.2 ข้อเสนอแนะ

- 1) เนื่องจากระบบถูกสร้างขึ้นใหม่ ในส่วนของหน้าเว็บบางส่วนยังออกแบบได้ไม่สมบูรณ์ ควรมีการแก้ไขเพิ่มเติมเพื่อให้เกิดความสวยงามและมีความสมบูรณ์มากยิ่งขึ้น
- 2) ในการร้องขอการปล่อยซอฟต์แวร์บางตัวยังไม่รองรับ เนื่องจากมีข้อจำกัดด้านสิทธิ์ในการอนุมัติ จำเป็นต้องมีการพัฒนาระบบต่อไปให้สามารถรองรับทุกๆ ซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

- [1] blog.sogoodweb.com. (2559). TypeScript ภาษาตัวแทน JavaScript จาก Microsoft. [Online]. Available: <https://blog.sogoodweb.com/Article/Detail/54455>. เข้าถึงเมื่อวันที่ 2 สิงหาคม 2561.
- [2] codebee. (2016). API คืออะไร ทำความรู้จักกับเอพีไอ. [Online]. Available: <https://www.codebee.co.th/labs/api-คืออะไร-ทำความรู้จัก/>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [3] codingbasic.com. (ไม่ปรากฏปีที่พิมพ์). CSS คืออะไร. [Online]. Available: <http://www.codingbasic.com/css.html>. เข้าถึงเมื่อวันที่ 2 สิงหาคม 2561.
- [4] devahoy.com. (2015). MongoDB คืออะไร? + สอนวิธีใช้งาน. [Online]. Available: <https://devahoy.com/posts/getting-started-with-mongodb/>เบื้องต้น. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [5] etda.or.th. (2014). จัดการ Windows Process ด้วย Windows Sysinternals Process Explorer. [Online]. Available: <https://www.etda.or.th/content/process-explorer.html>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [6] hostpacific.com. (2017). ทำความรู้จัก Docker และการใช้งานบน CentOS 7. [Online]. Available: <https://www.hostpacific.com/using-docker-on-centos7/>. เข้าถึงเมื่อวันที่ 3 สิงหาคม 2561.
- [7] khasathan.in.th. (2014). Web service หรือ Web API สำคัญกับการพัฒนา Web application และ Mobile application ยังไง?. [Online]. Available: <https://khasathan.in.th/archives/792>. เข้าถึงเมื่อวันที่ 2 สิงหาคม 2561.

[8] medium.com. (2017). Jenkins คืออะไร ?. [Online]. Available: <https://medium.com/@thanthaisriphum/jenkins-คืออะไร-49a6a66eda63>.  
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกรังห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
เข้าถึงเมื่อวันที่ 3 สิงหาคม 2561.

- [9] medium.com. (2017). **มาทดสอบ API ด้วย PostMan กันเถอะ.** [Online]. Available: <https://medium.com/@nest12345/มาทดสอบ-api-ด้วย-postman-กันเถอะ-c750a09c86ce>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [10] mindphp.com. (2560). **JavaScript คืออะไร จาวา สคริปต์ คือ ภาษาคอมพิวเตอร์ สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต.** [Online]. Available: <http://www.mindphp.com/คู่มือ/73-คืออะไร/2187-java-javascript-คืออะไร.html>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [11] mindphp.com. (2560). **html คืออะไร.** [Online]. Available: <http://mindphp.com/คู่มือ/73-คืออะไร/2026-html-คืออะไร.html>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [12] mindphp.com. (2560). **รู้จักกับ visual Studio Code (วิซวล สตูดิโอ โค้ด) โปรแกรมฟรีจากค่ายไมโครซอฟท์.** [Online]. Available: <https://www.mindphp.com/บทความ/microsoft/4829-visual-studio-code.html>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [13] oracle-java.blogspot. (2007). **สร้าง AJAX ด้วย dojo และ JSON.** [Online]. Available: <http://oracle-java.blogspot.com/2007/07/ajax-dojo-json.html>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [14] saixii.com. (2017). **RESTful หรือ REST คือ.** [Online]. Available: <https://saixiii.com/what-is-restful/>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [15] share.olanlab. (2017). **Node.js คืออะไร ?.** [Online]. Available: <http://share.olanlab.com/th/it/blog/view/271>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.
- [16] sysadmin.psu.ac.th. (2018). **รู้จัก Angular กันดีกว่า.** [Online]. Available: <https://sysadmin.psu.ac.th/2018/05/22/รู้จัก-angularjs-กันดีกว่า/>. เข้าถึงเมื่อวันที่ 1 สิงหาคม 2561.

เอกสารนี้เป็นเอกสารที่เผยแพร่สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้