



## รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบการจัดการบันทึกการลงเวลาการเข้า – ออกงาน

Time Attendance Management Systems

นายรชต ทองประศรี

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการสหกิจศึกษา ระบบการจัดการการลงเวลาการเข้า - ออกงาน

ชื่อ-สกุล นักศึกษา นายรชต ทองประศรี

คณะ วิศวกรรมศาสตร์

ภาควิชา วิศวกรรมคอมพิวเตอร์

ชื่อ-สกุล อาจารย์นิเทศ อาจารย์เกียรติณรงค์ ทองประเสริฐ

ชื่อ-สกุล ผู้นิเทศงาน คุณสุรศักดิ์ เจียมไพศาล

ชื่อสถานประกอบการ บริษัท ดาต้าโปรคอมพิวเตอร์ซิสเต็มส์ จำกัด

## บทคัดย่อ

ในองค์กรหรือบริษัทที่มีสาขาหรือสถานที่ทำงานหลายแห่ง มีพนักงานหรือลูกจ้างที่เข้า - ออกงานเป็นช่วงเวลา หรือต้องนำเวลาการทำงานมาคิดเป็นค่าจ้างให้พนักงาน มักประสบปัญหากับการที่ต้องตรวจสอบการเข้า - ออกเวลางานของพนักงานจำนวนมาก ซึ่งมีความยุ่งยากตามจำนวนของพนักงานและจำนวนสาขาขององค์กร อีกทั้งยังยากที่จะติดตามว่าพนักงานแต่ละคนเข้าทำงานตรงเวลาหรือได้บันทึกเวลาการเข้า - ออกงานตามความเป็นจริง จึงเกิดการนำความรู้ทางด้านเทคโนโลยีมาออกแบบ เพื่อให้ได้ระบบการจัดการบันทึกเวลาการเข้า - ออกงาน

ระบบการจัดการบันทึกเวลาการเข้า - ออกงานถูกออกแบบโดยใช้แนวคิดของสถาปัตยกรรมไมโครเซอร์วิสร่วมกับ RESTful API พัฒนาด้วยภาษาไพธอน ใช้ MongoDB เป็นฐานข้อมูล มีเว็บแอปพลิเคชันเป็นส่วนติดต่อกับผู้ใช้งาน ระบบเชื่อมต่อกับเครื่องบันทึกลายนิ้วมือจำนวนมากที่สามารถนำไปติดตั้งได้ตามสถานที่ทำงาน เพื่อตอบโจทยความต้องการของการใช้งานระบบ ซึ่งเมื่อพนักงานมาทำงานและบันทึกการเข้างานโดยการตรวจสอบลายนิ้วมือ ระบบจะทำการจัดเก็บบันทึกเวลาให้โดยอัตโนมัติ ผู้ใช้งานสามารถเรียกใช้ฟังก์ชันต่าง ๆ ผ่านเว็บแอปพลิเคชันเพื่อเรียกดูและจัดการข้อมูลได้ โดยการวิจัยครั้งนี้มีจุดประสงค์เพื่อออกแบบและพัฒนาระบบเพื่อนำไปใช้ทดแทนระบบเก่า เพิ่มประสิทธิภาพการทำงานให้แก่ระบบ และเพิ่มความสามารถด้านการรองรับความต้องการที่เพิ่มมากขึ้น

คำสำคัญ: สถาปัตยกรรมไมโครเซอร์วิส, RESTful API, เครื่องบันทึกลายนิ้วมือ

**Co-operative Title:** Time Attendance Management Systems

**Student Intern Name:** Rachata Thongprasri

**Faculty:** Engineering

**Department:** Computer Engineering

**Advisor Name:** Kiatnarong Tongprasert

**Mentor Name:** Surasak Jiampaisarn

**Company:** Datapro Computer Systems Co., Ltd

## ABSTRACT

In an organization or company that has many branches or workplaces, has employees or workers who work for a period or must use attendances to calculate wages for employees. They often encounter the problem about having a huge of attendances which increase according to the number of employees and branches. It is difficult to track whether each employee works on time or attendance has recorded with the truth. All of the above leads to Time Attendance Management Systems.

Time Attendance Management Systems or TAMS is designed by microservices and RESTful API concept, implementing by python, using MongoDB as database, using web application as user interface and connects to a fingerprint-scanners which are installed in the workplace to answer the requirements of system usage. When employees come to work and record an attendance through using fingerprint-scanner, systems will store an attendance in database automatically. User can use web application to view and manage a personal data or attendances. The research objective is to design and develop the system to replace the old one, to increase the performance of systems and upgrade scalability

**Keywords:** Microservices architecture, Restful API, Fingerprint-scanner

## กิตติกรรมประกาศ

ปริญญาานิพนธ์นี้เสร็จสมบูรณ์ได้ด้วยความช่วยเหลือจากหลายท่านทั้งในทางตรงและทางอ้อม ซึ่งผู้วิจัยจะทำให้สำเร็จไม่ได้หากปราศจากความช่วยเหลือจากบุคคลเหล่านี้

ขอบคุณบริษัท ดาต้าโปรดคอมพิวเตอรส์ซิสเต็มส์ จำกัด ที่ให้โอกาสผู้วิจัยได้เข้าไปศึกษาและเรียนรู้การทำงานในมุมมองเชิงธุรกิจ รวมถึงการใช้อุปกรณ์ต่าง ๆ และสถานที่ในการวิจัยโครงการนี้

ขอบคุณคุณสุรศักดิ์ เจียมไพศาล ผู้จัดการแผนก ESI/SIG/AIU ที่ให้คำปรึกษาและแนะนำหลักการที่ใช้ในการวิจัยตลอดตั้งแต่ต้นจนจบการวิจัย ซึ่งคำแนะนำทำให้ผู้วิจัยสามารถคลายความสงสัยและเข้าใจในเนื้อหาที่กำลังศึกษาได้มากยิ่งขึ้น รวมถึงการให้คำแนะนำด้านการทำงาน การแก้ไขปัญหา และจุดบกพร่องของการวิจัย

ขอบคุณพี่ทุกคนในแผนก ESI / SIG / AIU และพี่แผนกอื่น ๆ ที่ให้คำแนะนำด้านการทำงาน และการดูแลตลอดการฝึกงานและสหกิจศึกษา

ขอบคุณอาจารย์จิระศักดิ์ สิทธิกร และอาจารย์บัณฑิต พัสยา ซึ่งเป็นอาจารย์ผู้ดูแลและตอบคำถามเรื่องการฝึกงานและสหกิจศึกษาซึ่งทำให้กระบวนการของการฝึกงานและสหกิจศึกษาเป็นไปได้อย่างราบรื่นและสะดวก

ขอบคุณอาจารย์เกียรติณรงค์ ทองประเสริฐ อาจารย์ผู้มานิเทศงาน ที่ให้คำปรึกษาและคำแนะนำในการทำงานและแก้ไขปัญหา

สุดท้ายนี้ขอขอบคุณบิดา มารดา ครอบครัว เพื่อน ๆ ตลอดจนผู้ที่เกี่ยวข้องที่ไม่ได้กล่าวถึงทุกท่านที่เป็นกำลังใจ ให้การสนับสนุนและช่วยเหลือในการทำการวิจัยครั้งนี้จนสำเร็จลุล่วง

รชต ทองประศรี

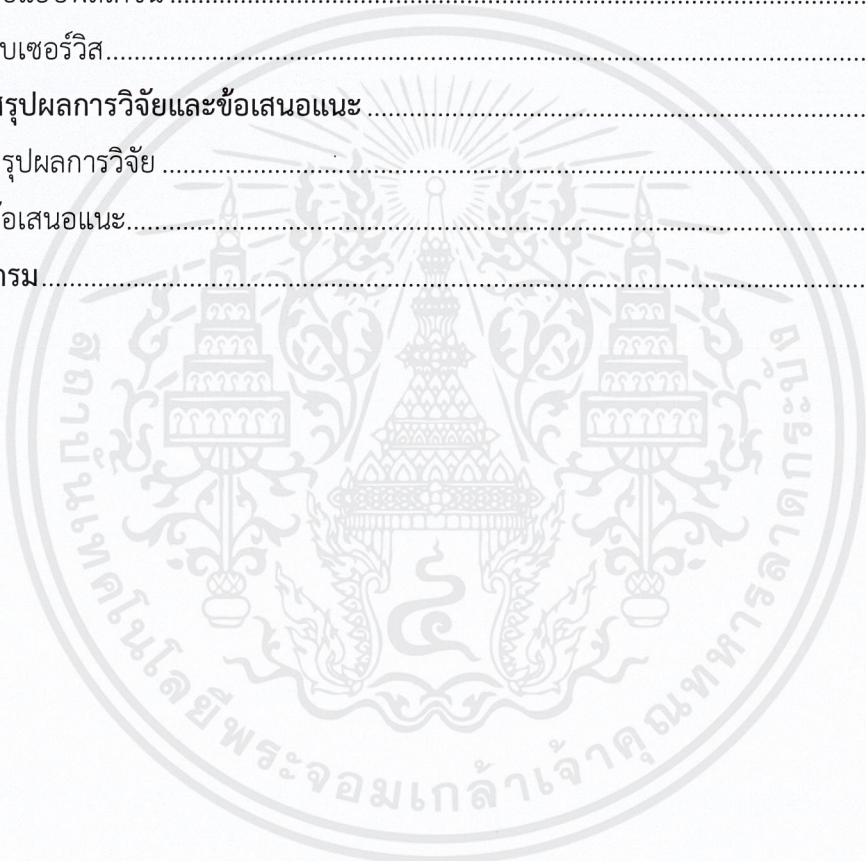
# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	1
1.3 ขอบเขตของการวิจัย.....	1
1.4 วิธีดำเนินการวิจัย.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ทฤษฎีที่เกี่ยวข้อง.....	3
บทที่ 3 วิธีการดำเนินการวิจัย.....	9
3.1 ภาพรวมของระบบ.....	9
3.2 ขั้นตอนการดำเนินงาน.....	9
3.3 การเก็บรวบรวมความต้องการ.....	10
3.4 การออกแบบไมโครเซอร์วิส (Microservices).....	11
3.4.1 ไมโครเซอร์วิสการจัดการข้อมูลผู้ใช้งาน (User).....	11
3.4.2 ไมโครเซอร์วิสการจัดการผู้ใช้งานในเครื่องอุปกรณ์ (Internal-user).....	11
3.4.3 ไมโครเซอร์วิสการจัดการบันทึกเวลาในฐานข้อมูล (Attendance).....	11
3.4.4 ไมโครเซอร์วิสการจัดการอุปกรณ์เครื่องบันทึกลายนิ้วมือ (Device).....	11
3.4.5 ไมโครเซอร์วิสการจัดการสถานที่ (Location).....	11
3.4.6 ไมโครเซอร์วิสส่วนของงานประจำ (Scheduler).....	11
3.4.7 ไมโครเซอร์วิสส่วนของการเก็บ Log ของโปรแกรม (Logging).....	12
3.4.8 ไมโครเซอร์วิสส่วนของการเก็บ URL.....	12
3.5 แผนภาพอธิบายโครงสร้างและการทำงาน.....	14
3.5.1 Use Case Diagrams.....	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและ IV อย่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
3.5.2 Flowchart.....	49
3.5.3 Entity Relationship Diagrams.....	62
3.5.4 Database Schema Diagram.....	62
3.6 การติดต่อสื่อสารกับเครื่องบันทึกถาดนิ้วมือ.....	65
<b>บทที่ 4 ผลการวิจัย.....</b>	<b>68</b>
4.1 เว็บไซต์พลิเคชัน.....	68
4.2 เว็บไซต์ervis.....	72
<b>บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....</b>	<b>75</b>
5.1 สรุปผลการวิจัย.....	75
5.2 ข้อเสนอแนะ.....	75
<b>บรรณานุกรม.....</b>	<b>76</b>



## สารบัญตาราง

ตารางที่	หน้า
3.1 Actor Employee .....	15
3.2 Actor HR.....	15
3.3 Actor Administrator .....	15
3.4 Actor Schedule.....	15
3.5 Login to web application .....	16
3.6 View personal information.....	16
3.7 Manage user.....	17
3.8 View user information.....	18
3.9 Edit user.....	19
3.10 Delete user .....	21
3.11 Enroll fingerprint.....	22
3.12 Clear user fingerprint .....	23
3.13 Create new user.....	24
3.14 View user attendance.....	25
3.15 Manage location .....	26
3.16 Create new location.....	27
3.17 View location .....	28
3.18 Edit location .....	29
3.19 Delete location .....	30
3.20 Manage device .....	31
3.21 Create new device .....	32
3.22 View device.....	33
3.23 Edit device .....	34
3.24 Delete device .....	35
3.25 Get user from database .....	36
3.26 Add new user to database .....	36
3.27 Update user to database .....	37
3.28 Save fingerprint to database .....	37
3.29 Clear user fingerprint from database.....	38

## สารบัญตาราง (ต่อ)

ตารางที่	หน้า
3.30 Add new user to device.....	38
3.31 Update user to device.....	39
3.32 Set fingerprint to device.....	39
3.33 Set device enrolling mode.....	40
3.34 Get fingerprint from device.....	41
3.35 Clear user fingerprint from device.....	41
3.36 Get user attendance from database.....	42
3.37 Add attendance to database.....	42
3.38 Set device time.....	43
3.39 Clear all attendance from device.....	43
3.40 Get attendance from device.....	44
3.41 Get device from database.....	45
3.42 Use case Add new device to database.....	45
3.43 Update device to database.....	46
3.44 Get location from database.....	47
3.45 Add new location to database.....	47
3.46 Update location to database.....	48
3.47 ตาราง User.....	63
3.48 ตาราง Attendance.....	63
3.49 ตาราง Location.....	64
3.50 ตาราง Device.....	64
3.51 ตาราง Log.....	64
3.52 ตาราง URL.....	64
3.53 ชุดคำสั่งที่ใช้ระบบ.....	66

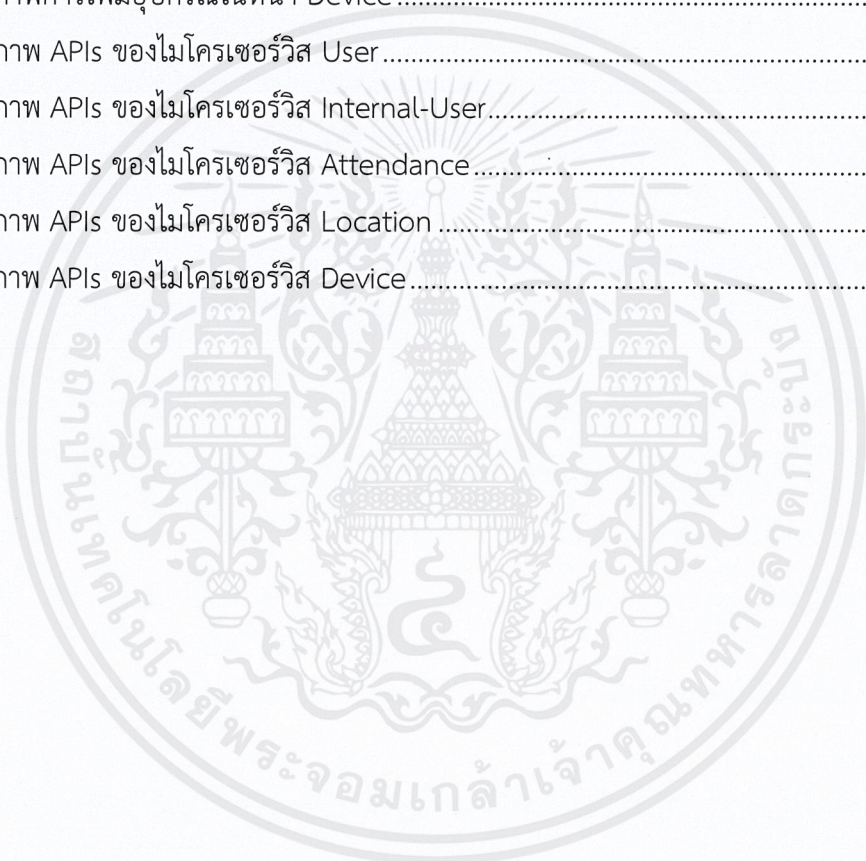
## สารบัญญภาพ

ภาพที่	หน้า
2.1 การเปรียบเทียบสถาปัตยกรรมโมโนลิธิกับไมโครเซอร์วิส.....	5
3.1 ภาพรวมการทำงานของระบบการจัดการบันทึกเวลาการเข้า - ออกงาน.....	12
3.2 สถาปัตยกรรมของระบบการจัดการบันทึกเวลาการเข้า - ออกงาน.....	13
3.3 แผนภาพยูสเคสของระบบการจัดการบันทึกเวลาการเข้า - ออกงาน.....	14
3.4 ฟังก์ชัน View personal information.....	49
3.5 ฟังก์ชัน View user information.....	49
3.6 ฟังก์ชัน Create new user.....	50
3.7 ฟังก์ชัน Edit user.....	51
3.8 ฟังก์ชัน Delete user.....	52
3.9 ฟังก์ชัน Enroll fingerprint.....	53
3.10 ฟังก์ชัน Clear fingerprint.....	54
3.11 ฟังก์ชัน View Location.....	55
3.12 ฟังก์ชัน Create new Location.....	55
3.13 ฟังก์ชัน Edit Location.....	56
3.14 ฟังก์ชัน Delete Location.....	57
3.15 ฟังก์ชัน View Device.....	58
3.16 ฟังก์ชัน Create new Device.....	58
3.17 ฟังก์ชัน Edit Device.....	59
3.18 ฟังก์ชัน Delete Device.....	60
3.19 ฟังก์ชัน Collecting Attendance.....	61
3.20 ความสัมพันธ์ระหว่างเอนทิตีของระบบการจัดการบันทึกเวลาเข้า - ออกงาน.....	62
3.21 แผนภาพ Database Schema ของระบบการจัดการบันทึกเวลาเข้า - ออกงาน.....	62
3.22 โครงสร้าง Packet ที่ใช้ในการสื่อสารกับเครื่องบันทึกเวลานี้ว่ามี.....	65
3.23 โพรโตคอล UDP ของการสื่อสารระหว่างอุปกรณ์กับผู้ที่ใช้.....	67
3.24 การสื่อสารระหว่างอุปกรณ์กับผู้ที่ใช้งาน.....	67
4.1 จอภาพหน้า Personal Information (Index).....	68
4.2 จอภาพหน้า User Attendance.....	68
4.3 จอภาพการเรียกดูข้อมูลบันทึกเวลาของพนักงานหน้า User Attendance.....	69
4.4 จอภาพหน้า User Management.....	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาแล VIII อย่างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4.5 จอภาพการเพิ่มพนักงานในหน้า User Management.....	69
4.6 จอภาพการแก้ไขข้อมูลพนักงานในหน้า User Management.....	70
4.7 จอภาพหน้า Location.....	70
4.8 จอภาพการเพิ่มสถานที่ในหน้า Location.....	70
4.9 จอภาพหน้า Device .....	71
4.10 จอภาพการเพิ่มอุปกรณ์ในหน้า Device .....	71
4.11 จอภาพ APIs ของไมโครเซอร์วิส User.....	72
4.12 จอภาพ APIs ของไมโครเซอร์วิส Internal-User.....	72
4.13 จอภาพ APIs ของไมโครเซอร์วิส Attendance.....	73
4.14 จอภาพ APIs ของไมโครเซอร์วิส Location .....	73
4.15 จอภาพ APIs ของไมโครเซอร์วิส Device.....	74



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญ

ในองค์กรหรือบริษัทที่มีสาขาหรือสถานที่ทำงานหลายแห่ง มีพนักงานหรือลูกจ้างที่เข้า - ออกงานเป็นช่วงเวลา หรือต้องนำเวลาการทำงานมาคิดเป็นค่าจ้างให้พนักงาน มักประสบปัญหาเกี่ยวกับการที่ต้องตรวจสอบการเข้า - ออกเวลางานของพนักงานจำนวนมาก ซึ่งมีความยุ่งยากตามจำนวนของพนักงานและจำนวนสาขาขององค์กร อีกทั้งยังยากที่จะติดตามว่าพนักงานแต่ละคนเข้าทำงานตรงเวลาหรือได้บันทึกเวลาการเข้า - ออกงานตามความเป็นจริง จึงเกิดการนำความรู้ทางด้านเทคโนโลยีมาออกแบบ เพื่อให้ได้ระบบการจัดการบันทึกเวลาการเข้า - ออกงาน

ระบบการจัดการบันทึกเวลาการเข้า - ออกงานถูกออกแบบและพัฒนาขึ้นเพื่อนำไปใช้ทดแทนระบบเก่าที่ใช้งานอยู่ เนื่องจากซอฟต์แวร์เก่านั้นมีขีดจำกัดในเรื่องประสิทธิภาพการทำงานเนื่องจากเป็นเทคโนโลยีเก่าซึ่งมีโอเวอร์เฮดสูง และติดปัญหาด้านการขยายเมื่อมีการใช้งานที่มากขึ้นซึ่งเมื่อองค์กรมีการรับพนักงานเพิ่มหรือการขยายสาขา การขยายของซอฟต์แวร์นั้นมีความจำเป็นอย่างมาก ผู้วิจัยจึงได้รับมอบหมายให้ออกแบบและพัฒนาาระบบนี้เพื่อให้ได้ระบบที่ใช้เทคโนโลยีที่ใหม่ขึ้น ลดโอเวอร์เฮดและเพิ่มประสิทธิภาพการทำงาน

### 1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อออกแบบและพัฒนาซอฟต์แวร์ทดแทนระบบเก่าด้วยเทคโนโลยีปัจจุบัน
2. เพื่อเพิ่มประสิทธิภาพการทำงานของซอฟต์แวร์
3. เพื่อเพิ่มความสามารถด้านการขยายเพื่อรองรับความต้องการที่เพิ่มมากขึ้นให้กับซอฟต์แวร์
4. เพื่อให้ผู้วิจัยเข้าใจแนวคิดและหลักการของสถาปัตยกรรมไมโครเซอร์วิส
5. เพื่อให้ผู้วิจัยได้ฝึกทักษะการออกแบบและพัฒนาซอฟต์แวร์
6. เพื่อให้ผู้วิจัยได้ฝึกทักษะการแก้ปัญหาที่พบบ่อยในการพัฒนาซอฟต์แวร์
7. เพื่อให้ผู้วิจัยได้นำความรู้ที่ได้ศึกษามาประยุกต์ใช้กับการพัฒนาซอฟต์แวร์

### 1.3 ขอบเขตของการวิจัย

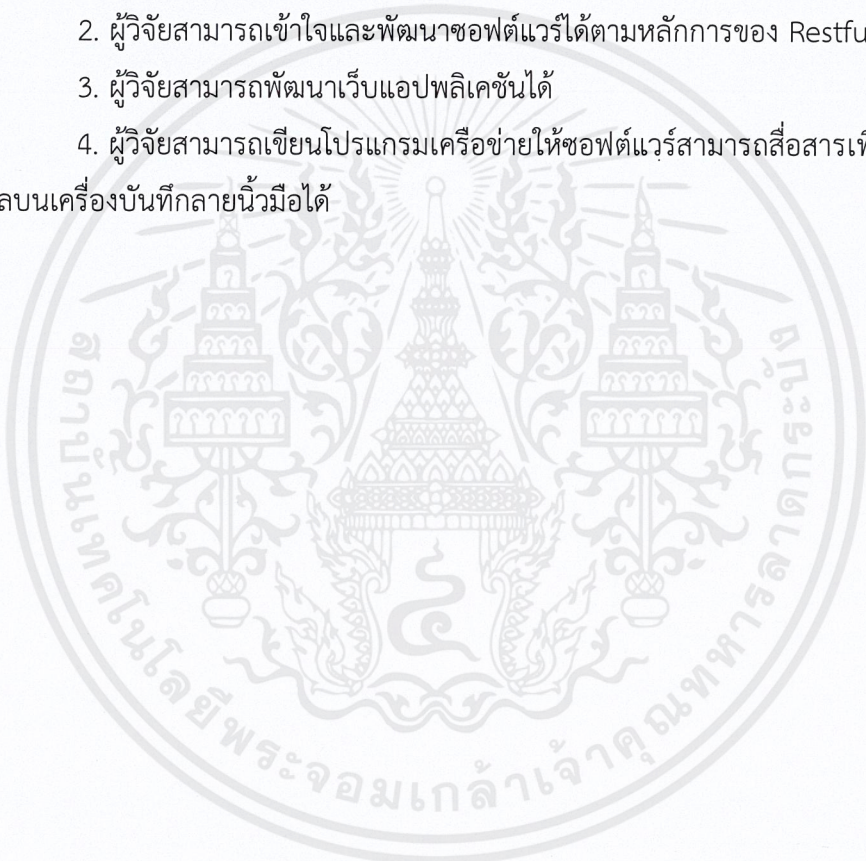
ซอฟต์แวร์ใหม่ที่ถูกออกแบบและพัฒนาขึ้นมาต้องใช้งานทดแทนซอฟต์แวร์เก่าได้ ผู้ใช้งานสามารถใช้ฟังก์ชันต่าง ๆ ของแอปพลิเคชันใหม่ได้เทียบเท่าแอปพลิเคชันเก่า การออกแบบซอฟต์แวร์ต้องเป็นไปตามสถาปัตยกรรมไมโครเซอร์วิสและ Restful API และซอฟต์แวร์ต้องมีประสิทธิภาพการทำงานที่ดีขึ้น

#### 1.4 วิธีดำเนินการวิจัย

1. วางแผนการดำเนินงานการวิจัย
2. ศึกษาความรู้ที่ต้องใช้ในงานวิจัย
3. ออกแบบและพัฒนาระบบ
4. ทดสอบระบบและแก้ไขปัญหา
5. สรุปผลการวิจัย

#### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ผู้วิจัยสามารถเข้าใจ ออกแบบและพัฒนาซอฟต์แวร์ได้อย่างถูกต้องตามหลักการของสถาปัตยกรรมไมโครเซอร์วิส
2. ผู้วิจัยสามารถเข้าใจและพัฒนาซอฟต์แวร์ได้ตามหลักการของ Restful API
3. ผู้วิจัยสามารถพัฒนาเว็บแอปพลิเคชันได้
4. ผู้วิจัยสามารถเขียนโปรแกรมเครือข่ายให้ซอฟต์แวร์สามารถสื่อสารเพื่ออ่านหรือเขียนข้อมูลบนเครื่องบันทึกถาวรได้



## บทที่ 2

### แนวคิดและทฤษฎีที่เกี่ยวข้อง

ในส่วนของแนวคิดและทฤษฎีที่เกี่ยวข้องนี้กล่าวถึงการนำแนวคิดและทฤษฎีในเรื่องต่าง ๆ ที่เกี่ยวข้องกับการวิจัยระบบการจัดการบันทึกเวลาการเข้า – ออกงาน ตั้งแต่การเริ่มออกแบบระบบจนถึงการพัฒนาซอฟต์แวร์

#### 2.1 ทฤษฎีที่เกี่ยวข้อง

##### 2.1.1 สถาปัตยกรรมเชิงบริการ

สถาปัตยกรรมเชิงบริการหรือ Service-oriented architecture (SOA) เป็นรูปแบบสถาปัตยกรรมการออกแบบและพัฒนาซอฟต์แวร์ที่ให้บริการแก่ซอฟต์แวร์ส่วนอื่น ๆ ผ่านทางโปรโตคอลการสื่อสารผ่านเครือข่าย หลักการพื้นฐานของสถาปัตยกรรมเชิงบริการคือเป็นอิสระทางด้านเทคโนโลยี โดยการบริการต่าง ๆ นั้นแยกเป็นหน่วยการทำงานที่สามารถเข้าถึงได้จากระยะไกลและสามารถดำเนินการหรือปรับปรุงแก้ไขได้อย่างอิสระ เช่น การเรียกใบแจ้งยอดบัตรเครดิตออนไลน์ การบริการมีคุณสมบัติ 4 ประการตามคำจำกัดความของสถาปัตยกรรมเชิงบริการดังนี้

1. การบริการสามารถแสดงถึงหน้าที่การทำงานได้อย่างมีเหตุผลพร้อมระบุผลลัพธ์ที่ออกมาได้อย่างชัดเจน
2. การบริการสามารถอยู่ด้วยตนเองได้
3. การบริการสามารถบอกผู้ใช้งานได้ว่าผู้ใช้งานมีลำดับการใช้งานอะไรไปบ้าง
4. การบริการอาจประกอบด้วยบริการพื้นฐานอื่น ๆ ได้

การบริการที่แตกต่างกันหลาย ๆ บริการสามารถใช้งานร่วมกันได้ในแอปพลิเคชันขนาดใหญ่ ซึ่งหลักการของสถาปัตยกรรมเชิงบริการนั้นใช้ร่วมกับการเขียนโปรแกรมแบบแยกส่วนได้ โดยสถาปัตยกรรมที่มุ่งเน้นการบริการจะรวมเอาส่วนประกอบต่าง ๆ ที่กระจายแยกส่วนกันของการเก็บและใช้งานของซอฟต์แวร์ส่วนต่าง ๆ เข้าด้วยกัน สถาปัตยกรรมนี้ถูกเปิดใช้งานโดยเทคโนโลยีและมาตรฐานที่อำนวยความสะดวกแก่การสื่อสารและการร่วมมือกันบนเครือข่าย โดยเฉพาะอย่างยิ่งบนเครือข่ายที่ใช้โปรโตคอลอินเทอร์เน็ต

สถาปัตยกรรมเชิงบริการนั้นมีการต่อยอดและปรับปรุงสถาปัตยกรรมไปในแนวทางต่าง ๆ โดยหนึ่งในสถาปัตยกรรมที่ถูกต่อยอดออกไปคือสถาปัตยกรรมไมโครเซอร์วิสที่จะกล่าวต่อไป

### 2.1.2 สถาปัตยกรรมไมโครเซอร์วิส

สถาปัตยกรรมไมโครเซอร์วิสเป็นการตีความที่ทันสมัยของสถาปัตยกรรมเชิงบริการที่ใช้เพื่อสร้างระบบของซอฟต์แวร์แบบกระจาย การบริการในสถาปัตยกรรมไมโครเซอร์วิสเป็นกระบวนการที่สื่อสารกันผ่านเครือข่ายเพื่อให้บรรลุเป้าหมายที่ต้องการ ไมโครเซอร์วิสเป็นแนวทางใหม่ในการนำสถาปัตยกรรมเชิงบริการไปใช้ คำจำกัดความของไมโครเซอร์วิสนั้นไม่แน่นอน โดยจะถูกปรับเปลี่ยนไปตามมุมมองของการพัฒนา โดยไมโครเซอร์วิสนั้นมีคุณสมบัติที่โดนอ้างถึงบ่อยเช่น

- การบริการในสถาปัตยกรรมไมโครเซอร์วิสสามารถใช้งานได้อย่างอิสระ
- แต่ละการบริการสามารถถูกพัฒนาโดยใช้ภาษาในการเขียนโปรแกรม

ฐานข้อมูล ซอฟต์แวร์ หรือฮาร์ดแวร์ที่แตกต่างกันได้ โดยสามารถเลือกใช้เทคโนโลยีที่เหมาะสมที่กับการบริการได้

- การบริการมีขนาดเล็ก มีหน้าที่เฉพาะของตัวเอง สามารถนำไปใช้ได้อย่างอิสระ สามารถกระจายและใช้งานได้ด้วยกระบวนการอัตโนมัติ

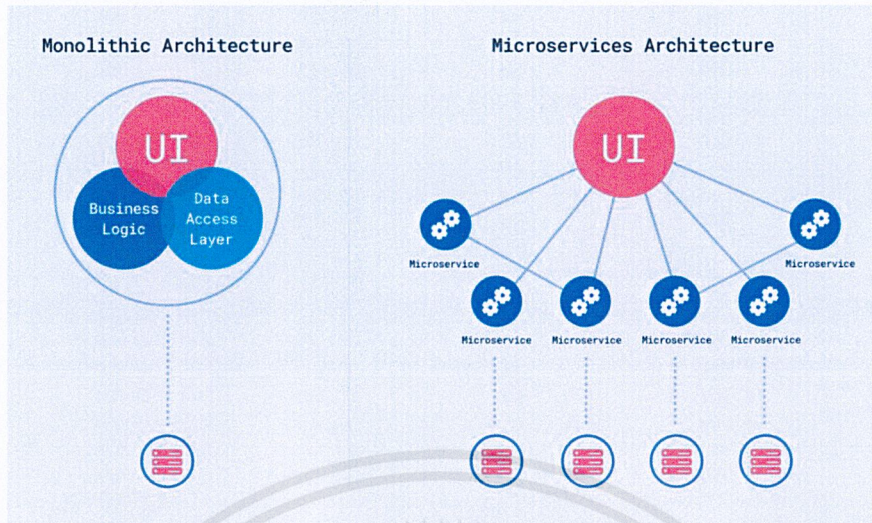
ผู้วิจัยได้รวบรวมคุณสมบัติของไมโครเซอร์วิสและจำกัดความไมโครเซอร์วิสว่า ไมโครเซอร์วิสเป็นรูปแบบสถาปัตยกรรมเชิงบริการ แต่ละไมโครเซอร์วิสจะแบ่งการบริการของระบบทั้งหมดให้เป็นส่วนที่มีหน้าที่เล็ก ๆ ที่ชัดเจนหลายบริการ โดยที่ภาพรวมของระบบยังมีการบริการที่ครบถ้วน ซึ่งแต่ละไมโครเซอร์วิสสามารถนำไปออกแบบหรือพัฒนาด้วยเทคนิคเทคโนโลยีหรือสภาพแวดล้อมที่แตกต่างกันได้และสามารถนำไปใช้งานได้อย่างอิสระ

ข้อดีของสถาปัตยกรรมไมโครเซอร์วิส

1. แต่ละไมโครเซอร์วิสมีอิสระในการเลือกใช้เทคโนโลยีในการพัฒนาที่แตกต่างกัน
2. เมื่อมีบริการในระบบเกิดข้อผิดพลาดที่ไม่สามารถใช้งานต่อได้ การบริการที่อยู่ในไมโครเซอร์วิสอื่นยังสามารถใช้งานได้ เนื่องจากบริการถูกแยกออกจากกันอย่างสมบูรณ์
3. สามารถจัดการและนำไมโครเซอร์วิสไปใช้งานได้ง่าย เนื่องจากไมโครเซอร์วิสมีขนาดเล็กและมีสภาพแวดล้อมเป็นของตนเอง
4. ช่วยประหยัดการใช้ทรัพยากรได้ในการขยายไมโครเซอร์วิสได้ เนื่องจากสามารถขยายบางไมโครเซอร์วิสที่ถูกใช้งานจำนวนมาก เพื่อการรองรับการใช้งานที่เพิ่มขึ้นโดยไม่จำเป็นต้องขยายในส่วนที่ไม่จำเป็น

อย่างไรก็ตามการออกแบบไมโครเซอร์วิสนั้นต้องใช้ความเข้าใจในตัวระบบที่กำลังออกแบบสูง ถ้าแบ่งบริการของระบบเป็นไมโครเซอร์วิสออกมาไม่มีประสิทธิภาพ การแก้ไขภายหลังจะเป็นเรื่องที่ยุ่งยาก การออกแบบไมโครเซอร์วิสก่อนการพัฒนาจึงมีความสำคัญอย่างมาก

## การเปรียบเทียบไมโครเซอร์วิสกับโมโนลิธ



ภาพที่ 2.1 การเปรียบเทียบสถาปัตยกรรมโมโนลิธกับไมโครเซอร์วิส

ที่มา : Alex Barashkov, 2561 : [https://dev.to/alex\\_barashkov/microservices-vs-monolith-architecture-4l1m?source=post\\_page-----c84d996d062-----](https://dev.to/alex_barashkov/microservices-vs-monolith-architecture-4l1m?source=post_page-----c84d996d062-----)

สถาปัตยกรรมโมโนลิธ (Monolith architecture) เป็นสถาปัตยกรรมเชิงบริการแบบหนึ่งที่ถูกใช้งานแพร่หลายในปัจจุบัน ซึ่งสถาปัตยกรรมโมโนลิธนั้นเป็นการออกแบบและพัฒนาซอฟต์แวร์ที่รวมส่วนต่าง ๆ ไว้ในสภาพแวดล้อมหรือโฮสต์เดียวกัน

การเปรียบเทียบคุณสมบัติระหว่างสถาปัตยกรรมโมโนลิธและไมโครเซอร์วิส

### 1. การติดตั้งระบบ (Deployment)

การติดตั้งของระบบที่เป็นโมโนลิธเขียนสคริปการติดตั้งที่จำเป็นเพียงครั้งเดียวที่ต้องครอบคลุมทั้งระบบ ส่วนการติดตั้งของระบบที่เป็นไมโครเซอร์วิสจะต้องเขียนสคริปสำหรับทุกไมโครเซอร์วิส แต่การเขียนสคริปแต่ละไมโครเซอร์วิสนั้นจะมีปริมาณและขนาดเล็กกว่าการเขียนให้กับระบบที่เป็นโมโนลิธ

### 2. การดูแลรักษา (Maintenance)

ระบบที่เป็นไมโครเซอร์วิสนั้นต้องการความรู้ด้านเครื่องมือต่าง ๆ เช่น Kubernetes, Docker Swarm หรือเครื่องมือที่คล้าย ๆ กัน

### 3. ความน่าเชื่อถือ (Reliability)

ระบบที่เป็นไมโครเซอร์วิสมีความน่าเชื่อถือที่สูงกว่าระบบของโมโนลิธ เนื่องจากเมื่อมีการพังของบริการ ระบบที่เป็นไมโครเซอร์วิสจะกระทบแค่บางส่วนของระบบ แต่ทางด้านของโมโนลิธนั้นมามารถใช้งานได้ทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4. ความสามารถในการขยาย (Scalability)

ระบบที่เป็นไมโครเซอร์วิสสามารถขยายได้ในส่วนที่ต้องการ หลีกเลี่ยงการใช้ทรัพยากรแบบไม่จำเป็นได้ดีกว่าของระบบที่เป็นโมโนลิธ นอกจากนี้ระบบที่เป็นไมโครเซอร์วิสยังสามารถขยายเพื่อรองรับความต้องการที่มากขึ้นเมื่อมีการใช้งานส่วนนั้นจำนวนมาก

#### 5. ต้นทุน (Cost)

เนื่องจากระบบที่เป็นโมโนลิธต้องคำนวณต้นทุนที่ต้องใช้ได้ถูกมากในบางสถานการณ์ แต่ในกรณีที่ต้องเพิ่มทรัพยากรเพื่อตอบโจทย์การขยายระบบนั้น ระบบที่เป็นโมโนลิธอาจจะต้องเจอกับค่าใช้จ่ายที่มหาศาล ต่างจากระบบที่เป็นไมโครเซอร์วิสซึ่งสามารถเลือกการขยายเป็น auto-scale เพื่อรับภาระต้นทุนในช่วงที่มีการใช้งานจำนวนมากเท่านั้น

#### 6. การพัฒนาระบบ (Development)

สมมติว่าระบบที่เป็นโมโนลิธสามารถออกแบบเป็นระบบที่เป็นไมโครเซอร์วิสได้ 13 ไมโครเซอร์วิส ในด้านของโมโนลิธต้องใช้ภาษาเดียวกันในการพัฒนาระบบ แต่ทางฝั่งไมโครเซอร์วิส ทั้ง 13 ไมโครเซอร์วิสนั้นสามารถใช้ภาษาที่แตกต่างกันได้ทั้งหมดตามความถนัดของกลุ่มของผู้พัฒนาที่ได้รับมอบหมายหน้าที่

ถ้ามองในมุมที่แคบลงมาในหนึ่งไมโครเซอร์วิสเปรียบเสมือนระบบโมโนลิธหนึ่งระบบที่มีขนาดเล็ก ภายในการออกแบบและพัฒนาหนึ่งระบบนี้สามารถใช้กระบวนการได้เหมือนการพัฒนาซอฟต์แวร์ทั่วไป อย่างไรก็ตามการออกแบบและเลือกใช้สถาปัตยกรรมนั้นเป็นส่วนหนึ่งของการวัดประสิทธิภาพของซอฟต์แวร์ ถ้าเลือกสถาปัตยกรรมที่เหมาะสมแต่การออกแบบและพัฒนาซอฟต์แวร์ไม่รัดกุมหรือการเขียนโปรแกรมที่ใช้ทรัพยากรมากเกินไปก็ส่งผลเสียต่อประสิทธิภาพของซอฟต์แวร์โดยตรง

### 2.1.3 สถาปัตยกรรม REST

สถาปัตยกรรม Representational State Transfer (REST) เป็นรูปแบบสถาปัตยกรรมซอฟต์แวร์ที่ใช้ประโยชน์จากเทคโนโลยีเว็บโพรโตคอลเพื่อใช้ในการสร้างเว็บเซอร์วิส เว็บเซอร์วิสที่สอดคล้องกับรูปแบบสถาปัตยกรรม REST เรียกว่า RESTful Web services ที่ให้การทำงานร่วมกันระหว่างระบบคอมพิวเตอร์บนอินเทอร์เน็ต รวมถึงการอนุญาตให้ระบบที่ร้องขอการเข้าถึงและการจัดการทรัพยากรของเว็บโดยการใช้ฟอร์มและชุดการดำเนินการที่ไม่เก็บสถานะที่กำหนดไว้ล่วงหน้า

ข้อกำหนดของคุณสมบัติทางสถาปัตยกรรม

1. ประสิทธิภาพในการโต้ตอบ
2. มีความสามารถในการปรับขยายได้ทำให้สามารถรองรับองค์ประกอบจำนวนมาก
3. ความเรียบง่ายของอินเตอร์เฟซที่เหมือนกัน
4. ความสามารถในการปรับเปลี่ยนส่วนประกอบเพื่อตอบสนองความต้องการที่เปลี่ยนแปลง แม้ในขณะที่แอปพลิเคชันทำงานอยู่

5. การมองเห็นการสื่อสารระหว่างองค์ประกอบผ่าน Service agents
6. ง่ายต่อการโยกย้ายในส่วนของโปรแกรมและข้อมูล
7. ความน่าเชื่อถือในการต้านทานความล้มเหลวในระบบ

ข้อจำกัดทางสถาปัตยกรรม

1. เป็นสถาปัตยกรรมโคลเอนต์ – เซิร์ฟเวอร์ โคลเอนต์ไม่มีหน้าที่เกี่ยวกับการจัดเก็บข้อมูล เซิร์ฟเวอร์มีหน้าที่เก็บทรัพยากรแต่ไม่จำเป็นต้องรู้สถานะผู้เรียกใช้
2. ไม่เก็บสถานะของการสื่อสารระหว่างโคลเอนต์กับเซิร์ฟเวอร์
3. โคลเอนต์สามารถเก็บแคชของกาตอบสนองจากเซิร์ฟเวอร์ได้ และการตอบสนองจะต้องกำหนดตัวเองว่าเป็นแคชหรือไม่ เพื่อป้องกันไม่ให้โคลเอนต์ได้รับข้อมูลเก่า
4. โคลเอนต์ไม่ทราบที่กำลังสื่อสารกับเซิร์ฟเวอร์กลางหรือเซิร์ฟเวอร์จริง
5. (มีหรือไม่มีก็ได้) เซิร์ฟเวอร์สามารถขยายหรือปรับแต่งการทำงานของโคลเอนต์ชั่วคราวโดยการโอนรหัสที่ปฏิบัติการได้
6. ข้อจำกัดของอินเตอร์เฟซที่เป็นแบบเดียวกันคือพื้นฐานในการออกแบบระบบ RESTful ใด ๆ ซึ่งเป็นการลดความซับซ้อนให้ระบบและแยกสถาปัตยกรรมของแต่ละระบบได้

REST เป็นกระบวนการทางสถาปัตยกรรม แต่ RESTful เป็นการใช้อ้างอิงถึงสถาปัตยกรรม นั้นหมายความว่า REST ไม่มีมาตรฐานเป็นของตัวเอง แต่ RESTful นั้นเป็นการใช้ประโยชน์จากเทคโนโลยีอื่น ๆ เช่น HTTP, URI, JSON และ XML นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.3 RESTful API

RESTful API เป็นส่วนการติดต่อระหว่างแอปพลิเคชันที่ใช้รูปแบบสถาปัตยกรรม REST ซึ่ง REST นั้นไม่มีมาตรฐานเป็นของตัวเอง แต่เป็นการใช้ประโยชน์จากเทคโนโลยีอื่น ๆ เพื่อการติดต่อสื่อสารระหว่างไคลเอนต์กับเซิร์ฟเวอร์โดยไม่เก็บสถานะการติดต่อเพื่อลดความซับซ้อน โดยการเรียก RESTful API ได้นั้นจำเป็นต้องเป็นไปตามข้อกำหนดและข้อจำกัดของ REST อีกด้วย

RESTful API กับ HTTP method

HTTP ได้กำหนดชุดคำสั่งซึ่งบ่งบอกชนิดของการกระทำที่จะกระทำกับทรัพยากรโดยในการวิจัยนี้ได้ใช้ HTTP method ในการร้องขอ (Request) ในการจัดการข้อมูลในทรัพยากรดังนี้

1. GET เป็นวิธีการสำหรับการร้องขอข้อมูลจากทรัพยากร
2. POST เป็นวิธีการสำหรับการสร้างข้อมูลใหม่ในทรัพยากร
3. PUT เป็นวิธีการสำหรับการเปลี่ยนแปลงข้อมูลที่อยู่ในทรัพยากรอยู่แล้ว
4. DELETE เป็นวิธีการสำหรับการลบข้อมูลในทรัพยากร

การร้องขอ (Request) จากไคลเอนต์ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์จะมีการตอบกลับการร้องขอ (Response) เสมอเพื่อให้ไคลเอนต์รู้ว่าการร้องขอนั้นสำเร็จหรือไม่ ถ้าไม่สำเร็จนั้นทำไมถึงไม่สำเร็จ โดยการอ้างอิงจากรหัสสถานะ HTTP ซึ่งเป็นรหัสมาตรฐานที่บ่งบอกสถานะการร้องขอที่ใช้กันโดยทั่วไป โดยรหัสสถานะ HTTP แบ่งออกเป็นหมวดดังนี้

หมวด Success - รหัส 2xx

- 200 OK เป็นมาตรฐานของการตอบกลับการร้องขอ HTTP ที่หมายความว่า การร้องขอ (GET, PUT, POST, DELETE) นั้นสำเร็จ

- 201 Create เป็นการตอบกลับการร้องขอสำหรับการสร้างข้อมูลใหม่

หมวด Client Error - รหัส 4xx

- 400 Bad Request บ่งบอกว่าเซิร์ฟเวอร์ไม่เข้าใจการร้องขอ

- 404 Not Found บ่งบอกว่าทรัพยากรที่ร้องขอมานั้นไม่วางใช้งานตอนนี้

หมวด Server Error - รหัส 5xx

- 500 Internal Server Error บ่งบอกว่าการร้องขอถูกต้อง แต่ฝั่งเซิร์ฟเวอร์เจอเหตุการณ์ไม่พึงประสงค์ในการให้บริการดังกล่าว และไม่สามารถให้บริการได้

ในแต่ละหมวดของรหัสสถานะ HTTP นั้นมีรหัสที่หลากหลาย แต่ผู้วิจัยยกเฉพาะรหัสที่ผู้ใช้งานพบบ่อยตลอดการวิจัยมาเรียบเรียง

## บทที่ 3

### วิธีการดำเนินการวิจัย

รายงานสหกิจฉบับนี้เป็นรายงานเกี่ยวกับระบบการจัดการบันทึกเวลาการเข้า – ออกงาน (Time Attendance Management Systems) ซึ่งบทนี้จะกล่าวถึงภาพรวมของระบบ ขั้นตอนการดำเนินงาน การเก็บรวบรวม วิเคราะห์ความต้องการ ออกแบบระบบ และการทำงานของระบบ

#### 3.1 ภาพรวมของระบบ

ระบบการจัดการบันทึกเวลาการเข้า – ออกงานถูกออกแบบโดยใช้สถาปัตยกรรมไมโครเซอร์วิส (Microservice Architecture) ซึ่งบริการ (Service) ต่าง ๆ จะถูกแบ่งออกเป็น ส่วน ๆ ซึ่งแต่ละไมโครเซอร์วิสถูกพัฒนาด้วยภาษาไพทอน (Python) โดยใช้ Flask-Restplus ที่เป็นส่วนขยายของ Flask framework ในการพัฒนาเพื่อช่วยในเรื่องการสร้างขึ้นเป็นเว็บแอปพลิเคชันที่ง่ายและสะดวก รวมถึงการทำ API document อีกด้วย

ด้านการติดต่อกับผู้ใช้งาน (User Interface) ผู้จัดทำเลือกใช้ Bootstrap และ Vue.js ในการพัฒนาเว็บแอปพลิเคชัน โดยเว็บแอปพลิเคชันนั้นแบ่งผู้ใช้งานเป็นระดับ ซึ่งแต่ละระดับจะเข้าถึงการใช้งานได้แตกต่างกัน จึงต้องมีการลงชื่อเข้าใช้เพื่อเข้าใช้งานได้อย่างถูกต้อง โดยมีระดับ

ระดับผู้ใช้งานทั่วไป (Employee) สามารถดูข้อมูลส่วนตัวของตนเองและบันทึกเวลาของตนเองได้เท่านั้น ระดับพนักงานฝ่ายบุคคลสามารถดูข้อมูล

ระดับพนักงานฝ่ายบุคคล (HR) สามารถเรียกดูข้อมูลต่าง ๆ ของพนักงานที่ต้องการและสามารถเพิ่ม แก้ไข จัดการข้อมูลส่วนตัวของพนักงาน รวมถึงการจัดการสถานที่ด้วย

ระดับผู้ดูแลระบบ (Administrator) สามารถจัดการข้อมูลของเครื่องบันทึกลายนิ้วมือในฐานะข้อมูลได้

ในส่วนฐานข้อมูลของไมโครเซอร์วิสต่าง ๆ ผู้วิจัยใช้ MongoDB ที่เป็นฐานข้อมูลประเภท NoSQL ในการจัดเก็บข้อมูลทั้งหมด

#### 3.2 ขั้นตอนการดำเนินงาน

1. เก็บรวบรวมความต้องการแอปพลิเคชัน
2. วิเคราะห์ความต้องการ และสรุปความคิดรวบยอดให้ตรงกับผู้นิเทศงาน
3. วางแผนการพัฒนาโครงการ
4. เลือกเทคโนโลยีที่จะใช้ในโครงการ
5. ศึกษาหัวข้อและรูปแบบต่าง ๆ ของเทคโนโลยีที่ใช้ในการพัฒนาโครงการ
6. ออกแบบโครงสร้างของระบบการจัดการบันทึกเวลาการเข้า – ออกงาน
7. พัฒนาระบบ ทดสอบและปรับปรุงการทำงานให้ถูกต้อง และมีประสิทธิภาพ
8. สรุปและจัดทำเอกสารอธิบายโครงสร้างและกระบวนการทำงานของโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การเก็บรวบรวมความต้องการ

จากการที่ผู้จัดทำได้เก็บรวบรวมความต้องการจากการพูดคุยระหว่างผู้นิเทศงานและ ผู้ที่มีประสบการณ์ในการวิเคราะห์และพัฒนาระบบการจัดการบันทึกเวลาการเข้า – ออกงาน จนเกิด เป็นคุณสมบัติของเว็บแอปพลิเคชันว่า

1. สามารถแสดงบันทึกเวลาการเข้า – ออกงานของพนักงานที่ต้องการได้
2. สามารถอ่านข้อมูลบันทึกเวลาจากเครื่องบันทึกเวลานี้ว่ามีทุกเครื่องที่เปิดใช้งาน มาบันทึกในฐานข้อมูลได้โดยอัตโนมัติ
3. สามารถเรียกดูข้อมูลส่วนบุคคลของพนักงานที่ต้องการจากฐานข้อมูลได้
4. สามารถกรอกข้อมูลเพื่อเพิ่มพนักงานไปยังฐานข้อมูลได้
5. สามารถแก้ไขข้อมูลต่าง ๆ ของพนักงานที่ต้องการได้
6. สามารถอ่านข้อมูลชื่อ นามสกุล รหัสประจำตัว และลายนิ้วมือของพนักงานจาก ฐานข้อมูลภายในเครื่องบันทึกเวลานี้ที่มีที่ต้องการ มาเก็บไว้ในฐานข้อมูลได้
7. สามารถสั่งการให้เครื่องบันทึกเวลานี้มีหรือเก็บลายนิ้วมือของพนักงานมาบันทึก ที่ฐานข้อมูลได้
8. สามารถเขียนข้อมูลชื่อ นามสกุล รหัสประจำตัว และลายนิ้วมือของพนักงานไปยัง เครื่องบันทึกเวลานี้ที่มีที่ต้องการได้
9. สามารถสร้างบัญชีผู้ใช้งานเว็บแอปพลิเคชันจากข้อมูลที่กรอกตอนลงทะเบียน พนักงานโดยอัตโนมัติ
10. สามารถแสดงข้อมูลต่าง ๆ ของเครื่องบันทึกเวลานี้ที่มีที่ต้องการได้
11. สามารถเพิ่มเครื่องบันทึกเวลานี้ที่มีไปยังฐานข้อมูลได้
12. สามารถแก้ไขข้อมูลเครื่องบันทึกเวลานี้ที่มีที่ต้องการได้
13. เครื่องบันทึกเวลานี้มีมีหลายกลุ่ม ซึ่งแยกอยู่ตามสถานที่ทำงานต่าง ๆ (อยู่ต่าง อาคาร หรือต่างสาขา) พนักงานสามารถลงเวลาได้เพียงที่ที่พนักงานทำงานอยู่เท่านั้น
14. ผู้ใช้งานสามารถ Login เข้าใช้เว็บแอปพลิเคชันซึ่งผู้ใช้งาน (User) นั้นมี 3 ระดับ ซึ่งจะใช้งานระบบนี้ได้แตกต่างกัน ดังนี้
  - 14.1) พนักงานทั่วไป – สามารถเรียกดูข้อมูลบันทึกเวลาของตนเองได้
  - 14.2) พนักงานฝ่ายบุคคล – สามารถเรียกดูข้อมูลบันทึกเวลาของพนักงาน อื่น ๆ ที่ต้องการได้ สามารถเรียกดูและจัดการข้อมูลพนักงานและข้อมูลของสถานที่ที่ต้องการได้
  - 14.3) พนักงานผู้ดูแล – สามารถจัดการข้อมูลของเครื่องบันทึกเวลานี้ที่มีได้
15. ต้องเก็บบันทึก Log การทำงานในส่วนที่จำเป็น
16. สามารถเรียกดู Log การทำงานของระบบเพื่อตรวจสอบข้อผิดพลาด และนำไป แก้ปัญหาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 การออกแบบไมโครเซอร์วิส (Microservices)

จากการเก็บรวบรวมและการวิเคราะห์ความต้องการ ผู้จัดทำได้ออกแบบระบบการจัดการบันทึกเวลาการเข้า – ออกงานตามสถาปัตยกรรมไมโครเซอร์วิส ดังนี้

#### 3.4.1 ไมโครเซอร์วิสการจัดการข้อมูลผู้ใช้งาน (User)

ส่วนของข้อมูลผู้ใช้งานเป็นส่วนที่มีฟังก์ชันในการจัดการข้อมูลของพนักงาน ประกอบด้วย การแสดงข้อมูลพนักงาน การเพิ่มข้อมูลพนักงานใหม่ การแก้ไขข้อมูลพนักงาน การลบข้อมูลพนักงาน โดยข้อมูลของพนักงานนั้นจะถูกจัดเก็บในฐานข้อมูล

#### 3.4.2 ไมโครเซอร์วิสการจัดการผู้ใช้งานในเครื่องอุปกรณ์ (Internal-user)

ส่วนของการจัดการผู้ใช้งานในเครื่องอุปกรณ์มีไว้สำหรับการติดต่อกับอุปกรณ์เครื่องบันทึกเวลานี้มีผู้อ่านข้อมูลต่าง ๆ และลายนิ้วมือของผู้ใช้งานภายในอุปกรณ์ (Internal device user) ออกมาเพื่อนำไปแสดงผลหรือเก็บในฐานข้อมูลข้างนอกอุปกรณ์ เขียนข้อมูลพนักงานไปบนอุปกรณ์ที่พนักงานนั้นต้องใช้งาน

#### 3.4.3 ไมโครเซอร์วิสการจัดการบันทึกเวลาในฐานข้อมูล (Attendance)

ส่วนของบันทึกเวลาในฐานข้อมูลเป็นส่วนที่มีไว้สำหรับการดึงข้อมูลบันทึกเวลาของพนักงานออกจากฐานข้อมูลเพื่อนำไปใช้งานตามเงื่อนไขที่ผู้ใช้งานต้องการ และสามารถบันทึกข้อมูลบันทึกเวลาเมื่อได้รับข้อมูลจากการอ่านบันทึกเวลาจากเครื่องบันทึกเวลานี้มี

#### 3.4.4 ไมโครเซอร์วิสการจัดการอุปกรณ์เครื่องบันทึกเวลานี้มีมือ (Device)

ส่วนของการจัดการอุปกรณ์เครื่องบันทึกเวลานี้มีมือเป็นส่วนที่เป็นตัวจัดการเรื่องอุปกรณ์เครื่องบันทึกเวลานี้มีมือทั้งหมด ทั้งเรื่องการเพิ่มอุปกรณ์ให้ระบบรู้จักและรับรู้ว่ามีอุปกรณ์นี้ทำงานอยู่บนระบบ การแก้ไขรายละเอียดข้อมูลของอุปกรณ์ การระงับการใช้งานอุปกรณ์ รวมถึงการตั้งค่าให้กับเครื่องบันทึกเวลานี้มีมือด้วย

#### 3.4.5 ไมโครเซอร์วิสการจัดการสถานที่ (Location)

สถานที่จะเป็นตัวบอกตำแหน่งของเครื่องบันทึกเวลานี้มีมือและสถานที่ทำงานของพนักงาน โดยส่วนนี้มีฟังก์ชันในการจัดการข้อมูลของสถานที่ การเพิ่มสถานที่ การแก้ไขข้อมูลสถานที่ และการลบสถานที่ รวมถึงการทำให้เครื่องบันทึกเวลานี้มีมือนั้นมีข้อมูลของพนักงานที่ต้องใช้งานอุปกรณ์เครื่องนั้น ๆ

#### 3.4.6 ไมโครเซอร์วิสส่วนของงานประจำ (Scheduler)

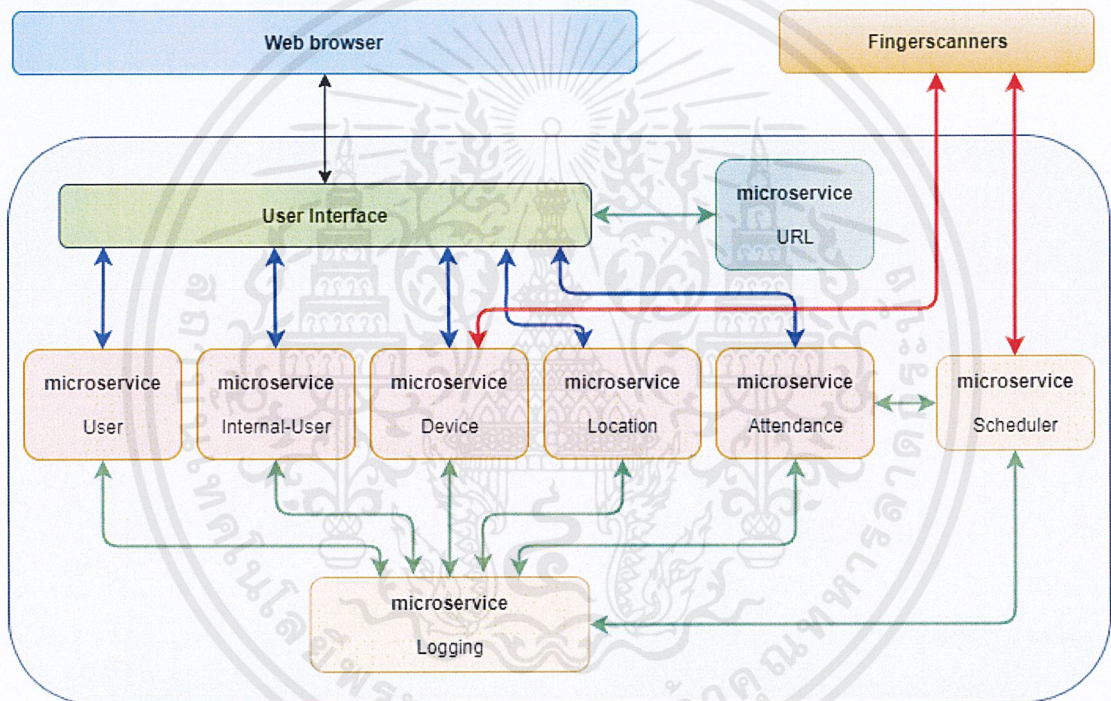
ส่วนนี้เป็นส่วนของงานประจำที่ระบบต้องทำเป็นตารางเวลา นั่นคือการเก็บบันทึกตารางเวลาจากอุปกรณ์ที่เปิดการใช้งานเพื่อนำข้อมูลบันทึกเวลาของพนักงานมาเก็บในฐานข้อมูล นอกจากนี้ยังมีการตั้งค่าเวลาให้กับอุปกรณ์ให้ตรงกับเวลาระบบ เพื่อให้ได้บันทึกเวลาที่ตรงกับเวลาจริง

### 3.4.7 ไมโครเซอร์วิสส่วนของการเก็บ Log ของโปรแกรม (Logging)

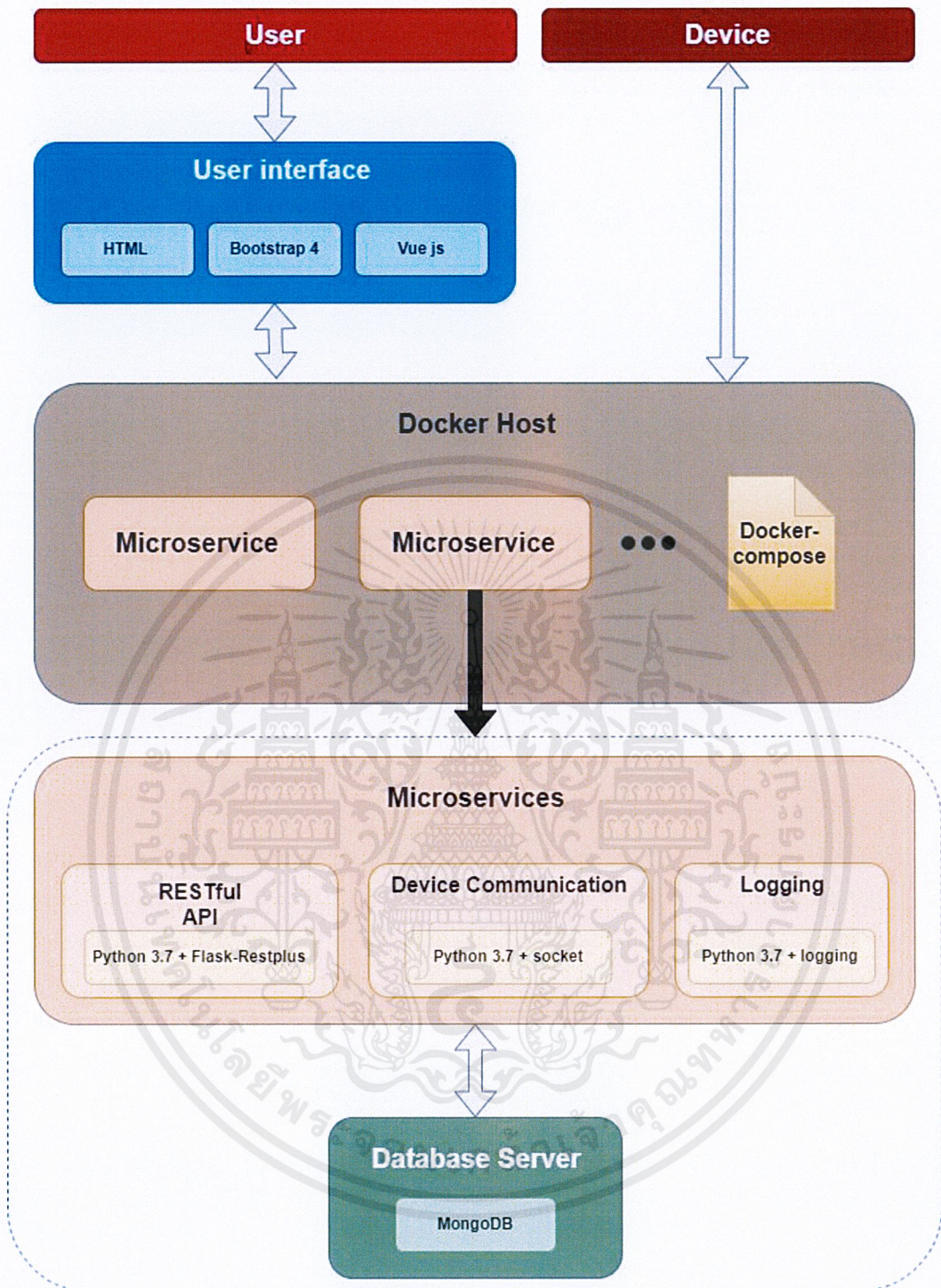
ส่วนนี้เป็นการรวบรวม Log การทำงานของฟังก์ชันในส่วนต่าง ๆ เพื่อการแก้ไขปัญหา (Troubleshooting) เมื่อมีการทำงานที่ผิดพลาดเกิดขึ้น โดยแนวคิดของส่วนนี้คือให้ฟังก์ชันในทุกส่วนนั้นส่งข้อมูลมาให้กับฟังก์ชันบันทึก Log ของส่วนนี้ เพื่อเป็นศูนย์กลางในการเก็บ Log ทั้งหมด

### 3.4.8 ไมโครเซอร์วิสส่วนของการเก็บ URL

ส่วนนี้เป็นการรวบรวม URL ของไมโครเซอร์วิสต่าง ๆ เพื่อการแก้ไขปัญหา เมื่อมีการทำงานที่ผิดพลาดเกิดขึ้น โดยแนวคิดของส่วนนี้คือให้ฟังก์ชันในทุกส่วนนั้นส่งข้อมูลมาให้กับฟังก์ชันบันทึก Log ของส่วนนี้ เพื่อเป็นศูนย์กลางในการเก็บ Log ทั้งหมด



ภาพที่ 3.1 ภาพรวมการทำงานของระบบการจัดการบันทึกเวลาการเข้า - ออกงาน



ภาพที่ 3.2 สถาปัตยกรรมของระบบการจัดการบันทึกเวลาการเข้า - ออกงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ผู้จัดทำได้อธิบายรายละเอียดของแผนภาพยูสเคสตามตารางที่ 3.1 - ตารางที่ 3.14

ตารางที่ 3.1 Actor Employee

Actor	Employee
ID	A1
Parent	-
Semantics: บุคคลที่เป็นพนักงานขององค์กรที่นำระบบนี้ไปใช้งาน	

ตารางที่ 3.2 Actor HR

Actor	HR
ID	A2
Parent	Employee
Semantics: บุคคลที่เป็นพนักงานฝ่ายบุคคลขององค์กรที่นำระบบนี้ไปใช้งาน	

ตารางที่ 3.3 Actor Administrator

Actor	Administrator
ID	A3
Parent	Employee
Semantics: บุคคลที่เป็นพนักงานผู้ดูแลระบบขององค์กรที่นำระบบนี้ไปใช้งาน	

ตารางที่ 3.4 Actor Schedule

Actor	Schedule
ID	A4
Parent	-
Semantics: ตัวกำหนดเวลาที่ต้องให้ระบบทำงานประจำ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.5 Login to web application

Use case	Login to web application
ID	UC1
Actors	Employee
Preconditions:	
<ol style="list-style-type: none"> <li>1. Employee เรียกใช้เว็บแอปพลิเคชัน</li> <li>2. Employee ยังไม่ได้เข้าสู่ระบบ</li> </ol>	
Flow of events:	
<ol style="list-style-type: none"> <li>1. เข้าหน้า Login ให้ Employee กรอกชื่อผู้ใช้และรหัสผ่าน</li> <li>2. Employee กรอกชื่อผู้ใช้และรหัสผ่านและกดปุ่ม Login</li> <li>3. ระบบจะทำการตรวจสอบบัญชีผู้ใช้งาน <ol style="list-style-type: none"> <li>3.1 ถ้าการเข้าสู่ระบบถูกต้อง ระบบจะให้ Employee เข้าสู่ระบบ</li> <li>3.2 ถ้าการเข้าสู่ระบบไม่ถูกต้อง ระบบจะคงอยู่หน้า Login และให้กรอกใหม่</li> </ol> </li> </ol>	
Postconditions:	
Employee เข้าใช้งานระบบ หน้าเว็บแสดงผลหน้า Personal information	

### ตารางที่ 3.6 View personal information

Use case	View personal information
ID	UC2
Actors	Employee
Preconditions:	
Employee เข้าสู่ระบบ	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. Employee เลือกปุ่ม User information ที่แถบซ้ายของหน้าเว็บในส่วนของ Personal Information</li> <li>2. ระบบเรียกหน้า Personal information (index)</li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>1. Get user from database</li> <li>2. Get user attendance from database</li> </ol>	
Postconditions:	
หน้าเว็บแสดงข้อมูลส่วนตัวของพนักงานและประวัติบันทึกเวลาของพนักงาน	

Secondary scenario: Get user from database ระบบ User Management ส่งข้อมูลส่วนตัวของ Employee ไปยังหน้าเว็บ
Postconditions of secondary scenario: Get user from database หน้าเว็บแสดงผลข้อมูลบันทึกเวลาของ Employee
Secondary scenario: Get user attendance from database ระบบ Attendance ส่งข้อมูลบันทึกเวลาของ Employee ไปยังหน้าเว็บ
Postconditions of secondary scenario: View device information หน้าเว็บแสดงผลข้อมูลบันทึกเวลาของ Employee

### ตารางที่ 3.7 Manage user

Use case	Manage user
ID	UC3
Actors	HR
Preconditions: HR เข้าสู่ระบบ	
Primary scenario: <ol style="list-style-type: none"> <li>HR เลือกปุ่ม Employee Management ที่แถบซ้ายของหน้าเว็บในส่วนของ User Information</li> <li>ระบบแสดง Drop down หรือ Pop up ขึ้นมาให้ HR เลือก</li> <li>HR เลือก Manage Employee ที่อยู่ในหัวข้อ Personal Information</li> <li>ระบบเรียกหน้า User Management</li> </ol>	
Secondary scenario: <ol style="list-style-type: none"> <li>Create new user</li> <li>View user information</li> </ol>	
Postconditions: หน้าเว็บแสดงปุ่ม + สำหรับเพิ่มพนักงานและข้อมูลส่วนตัวของพนักงานทั้งหมด	
Secondary scenario: Create new user HR กดปุ่ม + เพื่อเพิ่มผู้ใช้งาน	
Postconditions of secondary scenario: Create new user หน้าเว็บแสดง Drop down ภายในมีช่องให้กรอกข้อมูลของพนักงาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Secondary scenario: View user ระบบเรียกหน้า User Management
Postconditions of secondary scenario: View user หน้าเว็บแสดงผลข้อมูลส่วนตัวของพนักงาน

### ตารางที่ 3.8 View user information

Use case	View user information
ID	UC4
Actors	HR
Preconditions: <ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> </ol>	
Primary scenario: <ol style="list-style-type: none"> <li>1. ร้องขอข้อมูลพนักงานไปยัง User database</li> <li>2. แสดงข้อมูลของพนักงานที่ได้รับจากฐานข้อมูลบนหน้าเว็บ</li> <li>3. สร้างปุ่มให้กับข้อมูลของพนักงานแต่ละคนเพื่อเลือกแก้ไขข้อมูลพนักงานที่ต้องการได้</li> </ol>	
Secondary scenario: <ol style="list-style-type: none"> <li>1. Get user from database</li> <li>2. Edit user</li> <li>3. Delete user</li> <li>4. Enroll fingerprint</li> <li>5. Clear user fingerprint</li> </ol>	
Postconditions: หน้าเว็บแสดงผลข้อมูลของพนักงานทั้งหมด	
Secondary scenario: Get user from database ร้องขอข้อมูลของพนักงานทั้งหมดไปยัง User database	
Postconditions of secondary scenario: Get user from database ระบบ User Management ส่งข้อมูลของพนักงานไปยังส่วนที่เรียกใช้	
Secondary scenario: Edit user HR เลือกพนักงานที่ต้องการแก้ไขข้อมูล	
Postconditions of secondary scenario: Edit user หน้าเว็บแสดงผลหน้าตาส่วนของช่องให้กรอกข้อมูลของพนักงาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Secondary scenario: Delete user HR เลือกพนักงานที่ต้องการลบและเลือกปุ่มรูปถังขยะ
Postconditions of secondary scenario: Delete user ข้อมูลพนักงานถูกลบและเรียกหน้า User Management ใหม่
Secondary scenario: Enroll fingerprint HR เลือกพนักงานที่ต้องการเก็บลายนิ้วมือ
Postconditions of secondary scenario: Enroll fingerprint ข้อมูลลายนิ้วมือของพนักงานถูกเก็บลงในฐานข้อมูล
Secondary scenario: Clear user fingerprint HR เลือกพนักงานที่ต้องการลบลายนิ้วมือ
Postconditions of secondary scenario: Clear user fingerprint ลายนิ้วมือของพนักงานถูกลบออก

### ตารางที่ 3.9 Edit user

Use case	Edit user
ID	UC5
Actors	HR
Preconditions:	<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> <li>3. HR เลือกพนักงานที่ต้องการแก้ไขข้อมูล</li> </ol>
Primary scenario:	<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR กรอกข้อมูลใหม่และกด OK <ol style="list-style-type: none"> <li>2.1 ถ้าบันทึกข้อมูลใหม่สำเร็จ เรียกหน้า User Management ใหม่</li> <li>2.3 ถ้าบันทึกข้อมูลใหม่ไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>
Secondary scenario:	<ol style="list-style-type: none"> <li>1. Update user to database</li> <li>2. Update user to device</li> <li>3. Set fingerprint to device</li> <li>4. Clear fingerprint</li> </ol>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Postconditions: ข้อมูลใหม่ของพนักงานที่ HR แก้ไขถูกบันทึกในฐานข้อมูลและหน้าเว็บแสดงผลข้อมูลใหม่
Secondary scenario: Update user to database ระบบ User Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล
Postconditions of secondary scenario: Update user to database ข้อมูลพนักงานในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ
Secondary scenario: Update user to device ระบบ Internal-User Management แก้ไขข้อมูลของพนักงานในเครื่องบันทึกลายนิ้วมือ
Postconditions of secondary scenario: Update user to device ข้อมูลพนักงานในเครื่องบันทึกลายนิ้วมือถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ
Secondary scenario: Set fingerprint to device HR เปลี่ยนข้อมูลสถานที่ของพนักงาน
Postconditions of secondary scenario: Set fingerprint to device เพิ่มลายนิ้วมือไปยังเครื่องบันทึกลายนิ้วมือของสถานที่ใหม่
Secondary scenario: Clear fingerprint HR เปลี่ยนข้อมูลสถานที่ของพนักงาน
Postconditions of secondary scenario: Clear fingerprint ลายนิ้วมือถูกลบจากเครื่องบันทึกลายนิ้วมือของสถานที่เก่า

## ตารางที่ 3.10 Delete user

Use case	Delete user
ID	UC6
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> <li>3. HR เลือกพนักงานที่ต้องการลบออกจากระบบ</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR เลือกปุ่มลบในส่วนการแก้ไขข้อมูล <ol style="list-style-type: none"> <li>2.1 ถ้าลบพนักงานสำเร็จ เรียกหน้า User Management ใหม่</li> <li>2.3 ถ้าลบพนักงานไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>1. Update user to database</li> <li>2. Update user to device</li> <li>3. Clear fingerprint</li> </ol>	
Postconditions:	
ข้อมูลของพนักงานถูกลบออกจากระบบ	
Secondary scenario: Update user to database	
ระบบ User Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Update user to database	
ข้อมูลพนักงานในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	
Secondary scenario: Update user to device	
ระบบ Internal-User Management แก้ไขข้อมูลของพนักงานในเครื่องบันทึกลายนิ้วมือ	
Postconditions of secondary scenario: Update user to device	
ข้อมูลพนักงานในเครื่องบันทึกลายนิ้วมือถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	
Secondary scenario: Clear fingerprint	
ลบลายนิ้วมือของพนักงานที่เลือกออกจากฐานข้อมูลและเครื่องบันทึกลายนิ้วมือทั้งหมด	
Postconditions of secondary scenario: Clear fingerprint	
ลายนิ้วมือของพนักงานถูกลบออกจากระบบ	

### ตารางที่ 3.11 Enroll fingerprint

Use case	Enroll fingerprint
ID	UC7
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> <li>3. HR เลือกพนักงานที่ต้องการเก็บลายนิ้วมือ</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR เลือกปุ่ม Enroll ในส่วนการแก้ไขข้อมูล ระบบส่งคำสั่งไปยังเครื่องบันทึกลายนิ้วมือให้รออ่านลายนิ้วมือ</li> </ol>	
Secondary scenario:	
Set device enrolling mode	
Postconditions:	
ลายนิ้วมือของพนักงานถูกเก็บในฐานข้อมูล	
Secondary scenario: Set device enrolling mode	
ระบบ Internal-User Management สั่งให้เครื่องบันทึกลายนิ้วมือรออ่านลายนิ้วมือของพนักงาน	
Postconditions of secondary scenario: Set device enrolling mode	
ข้อมูลลายนิ้วมือถูกเก็บลงในฐานข้อมูล	

### ตารางที่ 3.12 Clear user fingerprint

Use case	Clear user fingerprint
ID	UC8
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> <li>3. HR เลือกพนักงานที่ต้องการลบลายนิ้วมือของพนักงานออกจากระบบ</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR เลือกปุ่ม Clear fingerprint ในส่วนการแก้ไขข้อมูล</li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>1. Clear user fingerprint from database</li> <li>2. Clear user fingerprint from device</li> </ol>	
Postconditions:	
ลายนิ้วมือของพนักงานถูกลบออกจากระบบ	
Secondary scenario: Clear user fingerprint from database	
ระบบ User Management ลบลายนิ้วมือของพนักงานทั้งหมดออกจากฐานข้อมูล	
Postconditions of secondary scenario: Clear user fingerprint from database	
ลายนิ้วมือของพนักงานถูกลบออกจากฐานข้อมูล	
Secondary scenario: Clear user fingerprint from device	
ระบบ Internal-User Management ลบลายนิ้วมือของพนักงานทั้งหมดออกจากเครื่องบันทึกลายนิ้วมือทั้งหมด	
Postconditions of secondary scenario: Clear user fingerprint from device	
ลายนิ้วมือของพนักงานถูกลบออกจากเครื่องบันทึกลายนิ้วมือทั้งหมด	

ตารางที่ 3.13 Create new user

Use case	Create new user
ID	UC9
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า User Management</li> <li>3. HR เลือกปุ่ม + เพื่อเพิ่มพนักงาน</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการเพิ่มข้อมูล</li> <li>2. HR กรอกข้อมูลพนักงานใหม่และกด Create new user <ol style="list-style-type: none"> <li>2.1 ถ้าเพิ่มพนักงานใหม่สำเร็จ เรียกหน้าเพจ User Management ใหม่</li> <li>2.2 ถ้าเพิ่มพนักงานไม่สำเร็จ หน้าเพจจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>1. Add new user to database</li> <li>2. Add new user to device</li> </ol>	
Postconditions:	
หน้าเว็บเรียกหน้า User Management ใหม่อัตโนมัติ	
Secondary scenario: Add new user to database	
ระบบ User Management รับข้อมูลที่ HR กรอกไปบันทึกในฐานข้อมูล	
Postconditions of secondary scenario: Add new user to database	
ส่งสถานะการบันทึกฐานข้อมูลไปยังหน้าเว็บ ถ้าบันทึกไม่สำเร็จจะส่งคำอธิบายไปด้วย	
Secondary scenario: Add new user to device	
ระบบ Internal-User Management รับข้อมูลที่ HR กรอกไปบันทึกในอุปกรณ์เครื่องบันทึกถายนิ้วมือ	
Postconditions of secondary scenario: Add new user to device	
ส่งสถานะการบันทึกเครื่องบันทึกถายนิ้วมือไปยังหน้าเว็บ ถ้าบันทึกไม่สำเร็จจะส่งคำอธิบายไปด้วย	

ตารางที่ 3.14 View user attendance

Use case	View user attendance
ID	UC10
Actors	HR
Preconditions: HR เข้าสู่ระบบ	
Primary scenario: 1. HR เลือกปุ่ม Attendance ที่ แถบซ้ายของหน้าเว็บในส่วนของ Attendance Management 2. ระบบแสดง Drop down หรือ Pop up ขึ้นมาให้ HR เลือก 3. HR เลือกตัวเลือก User Attendance เพื่อดูข้อมูลของพนักงานที่ HR ต้องการ 4. ระบบแสดงหน้าเว็บ User Attendance ให้ HR กรอกรหัสพนักงานเพื่อดูข้อมูล 5. HR กด Search for attendance	
Secondary scenario: View personal information	
Postconditions: หน้าเว็บแสดงข้อมูลส่วนตัวและประวัติบันทึกเวลาของรหัสพนักงานที่ HR กรอก	
Secondary scenario: View personal information เรียกดูข้อมูลส่วนตัวและประวัติบันทึกเวลาของพนักงาน	
Postconditions of secondary scenario: View personal information หน้าเว็บแสดงข้อมูลส่วนตัวของพนักงานและประวัติบันทึกเวลาของพนักงาน	

### ตารางที่ 3.15 Manage location

Use case	Manage location
ID	UC11
Actors	HR
Preconditions: HR เข้าสู่ระบบ	
Primary scenario: 1. HR เลือกปุ่ม Employee Management ที่แถบซ้ายของหน้าเว็บในส่วนของ User Information 2. ระบบแสดง Drop down หรือ Pop up ขึ้นมาให้ HR เลือกตัวเลือก 3. HR เลือก Manage Location ที่อยู่ในหัวข้อ Location 4. ระบบเรียกหน้า Location	
Secondary scenario: 1. Create new location 2. View location	
Postconditions: หน้าเว็บแสดงปุ่ม + สำหรับเพิ่มสถานที่และแสดงข้อมูลสถานที่ทั้งหมด	
Secondary scenario: Create new location HR กดปุ่ม + เพื่อเพิ่มสถานที่	
Postconditions of secondary scenario: Create new location หน้าเว็บแสดงหน้าต่างที่มีช่องให้กรอกข้อมูลของสถานที่	
Secondary scenario: View location ระบบเรียกหน้า Location	
Postconditions of secondary scenario: View location หน้าเว็บแสดงผลข้อมูลสถานที่ทั้งหมด	

ตารางที่ 3.16 Create new location

Use case	Create new location
ID	UC12
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า Location</li> <li>3. HR เลือกปุ่ม + เพื่อเพิ่มสถานที่</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการเพิ่มข้อมูล</li> <li>2. HR กรอกข้อมูลสถานที่ใหม่และกด Create new location               <ol style="list-style-type: none"> <li>2.1 ถ้าเพิ่มสถานที่ใหม่สำเร็จ เรียกหน้าเว็บ Location ใหม่</li> <li>2.2 ถ้าเพิ่มสถานที่ไม่สำเร็จ หน้าเพจจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Add new location to database	
Postconditions:	
สถานที่ใหม่ถูกเพิ่มในฐานข้อมูลและเรียกหน้า Location ใหม่	
Secondary scenario: Add new location to database	
ระบบ Location Management รับข้อมูลสถานที่ที่ HR กรอกไปบันทึกในฐานข้อมูล	
Postconditions of secondary scenario: Add new location to database	
ส่งสถานะการบันทึกลงฐานข้อมูลไปยังหน้าเว็บ ถ้าบันทึกไม่สำเร็จจะส่งคำอธิบายไปด้วย	

ตารางที่ 3.17 View location

Use case	View location
ID	UC13
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า Location</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. ร้องขอข้อมูลสถานที่ไปยัง Location database</li> <li>2. แสดงข้อมูลของสถานที่ที่ได้รับจากฐานข้อมูลบนหน้าเว็บ</li> <li>3. สร้างปุ่มให้กับข้อมูลของสถานที่แต่ละแห่งเพื่อเลือกแก้ไขข้อมูลสถานที่ที่ต้องการได้</li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>1. Get location from database</li> <li>2. Edit location</li> <li>3. Delete location</li> </ol>	
Postconditions:	
หน้าเว็บแสดงข้อมูลของสถานที่	
Secondary scenario: Get location from database	
ร้องขอข้อมูลของสถานที่ทั้งหมดไปยัง Location database	
Postconditions of secondary scenario: Get location from database	
ระบบ Location Management ส่งข้อมูลของสถานที่ไปยังส่วนที่เรียกใช้	
Secondary scenario: Edit location	
HR เลือกสถานที่ที่ต้องการแก้ไขข้อมูล	
Postconditions of secondary scenario: Edit location	
หน้าเว็บแสดงผลหน้าตาบางส่วนของช่องให้กรอกข้อมูลของสถานที่	
Secondary scenario: Delete location	
HR เลือกสถานที่ที่ต้องการลบและเลือกปุ่มรูปถังขยะ	
Postconditions of secondary scenario: Delete location	
ข้อมูลสถานที่ถูกลบและเรียกหน้า Location ใหม่	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.18 Edit location

Use case	Edit location
ID	UC14
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า Location</li> <li>3. HR เลือกสถานที่ที่ต้องการแก้ไขข้อมูล</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR กรอกข้อมูลใหม่และกด OK <ol style="list-style-type: none"> <li>2.1 ถ้าบันทึกข้อมูลใหม่สำเร็จ เรียกหน้า Location ใหม่</li> <li>2.3 ถ้าบันทึกข้อมูลใหม่ไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Update location to database	
Postconditions:	
ข้อมูลใหม่ของสถานที่ที่ HR แก้ไขถูกบันทึกในฐานข้อมูลและหน้าเว็บแสดงผลข้อมูลใหม่	
Secondary scenario: Update location to database	
ระบบ Location Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Update location to database	
ข้อมูลสถานที่ในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	

ตารางที่ 3.19 Delete location

Use case	Delete location
ID	UC15
Actors	HR
Preconditions:	
<ol style="list-style-type: none"> <li>1. HR เข้าสู่ระบบ</li> <li>2. HR เรียกใช้หน้า Location</li> <li>3. HR เลือกพนักงานที่ต้องการลบออกจากระบบ</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. HR เลือกปุ่มลบในส่วนการแก้ไขข้อมูล เพื่อลบข้อมูลสถานที่ <ol style="list-style-type: none"> <li>2.1 ถ้าลบสถานที่สำเร็จ เรียกหน้า Location ใหม่</li> <li>2.3 ถ้าลบสถานที่ไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Update location to database	
Postconditions:	
ข้อมูลของพนักงานถูกลบออกจากระบบ	
Secondary scenario: Update location to database	
ระบบ Location Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Update location to database	
ข้อมูลสถานที่ในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	

## ตารางที่ 3.20 Manage device

Use case	Manage device
ID	UC16
Actors	Administrator
Preconditions: Administrator เข้าสู่ระบบ	
Primary scenario: 1. Administrator เลือกปุ่ม Device Management ที่แถบซ้ายของหน้าเว็บในส่วนของ Device Management 2. ระบบแสดง Drop down หรือ Pop up ขึ้นมาให้ Administrator เลือกตัวเลือก 3. Administrator เลือก Manage device 4. ระบบเรียกหน้า Device Management	
Secondary scenario: 1. Create new device 2. View device	
Postconditions: หน้าเว็บแสดงปุ่ม + สำหรับเพิ่มอุปกรณ์และแสดงข้อมูลอุปกรณ์ทั้งหมดทั้งหมด	
Secondary scenario: Create new device Administrator กดปุ่ม + เพื่อเพิ่มอุปกรณ์	
Postconditions of secondary scenario: Create new device หน้าเว็บแสดงหน้าต่างที่มีช่องให้กรอกข้อมูลของอุปกรณ์	
Secondary scenario: View device ระบบเรียกหน้า Device Management	
Postconditions of secondary scenario: View device หน้าเว็บแสดงผลข้อมูลอุปกรณ์ทั้งหมด	

## ตารางที่ 3.21 Create new device

Use case	Create new device
ID	UC17
Actors	Administrator
Preconditions:	
<ol style="list-style-type: none"> <li>1. Administrator เข้าสู่ระบบ</li> <li>2. Administrator เรียกใช้หน้า Device Management</li> <li>3. Administrator เลือกปุ่ม + เพื่อเพิ่มอุปกรณ์</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการเพิ่มข้อมูล</li> <li>2. Administrator กรอกข้อมูลอุปกรณ์ใหม่และกด Create new device <ol style="list-style-type: none"> <li>2.1 ถ้าเพิ่มอุปกรณ์ใหม่สำเร็จ เรียกหน้าเว็บ Device Management ใหม่</li> <li>2.2 ถ้าเพิ่มอุปกรณ์ไม่สำเร็จ หน้าเพจจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Add new device to database	
Postconditions:	
อุปกรณ์ใหม่ถูกเพิ่มในฐานข้อมูลและเรียกหน้า Device Management ใหม่	
Secondary scenario: Add new device to database	
ระบบ Device Management รับข้อมูลอุปกรณ์ที่ Administrator กรอกไปบันทึกในฐานข้อมูล	
Postconditions of secondary scenario: Add new device to database	
ส่งสถานะการบันทึกฐานข้อมูลไปยังหน้าเว็บ ถ้าบันทึกไม่สำเร็จจะส่งคำอธิบายไปด้วย	

## ตารางที่ 3.22 View device

Use case	View device
ID	UC18
Actors	Administrator
Preconditions:	
<ol style="list-style-type: none"> <li>Administrator เข้าสู่ระบบ</li> <li>Administrator เรียกใช้หน้า Device Management</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>ร้องขอข้อมูลอุปกรณ์ไปยัง Device database</li> <li>แสดงข้อมูลของอุปกรณ์ที่ได้รับจากฐานข้อมูลบนหน้าเว็บ</li> <li>สร้างปุ่มให้กับข้อมูลของแต่ละอุปกรณ์เพื่อเลือกแก้ไขข้อมูลอุปกรณ์ที่ต้องการได้</li> </ol>	
Secondary scenario:	
<ol style="list-style-type: none"> <li>Get device from database</li> <li>Edit device</li> <li>Delete device</li> </ol>	
Postconditions:	
หน้าเว็บแสดงข้อมูลของอุปกรณ์	
Secondary scenario: Get device from database	
ร้องขอข้อมูลของสถานที่ทั้งหมดไปยัง Device database	
Postconditions of secondary scenario: Get device from database	
ระบบ Device Management ส่งข้อมูลของอุปกรณ์ไปยังส่วนที่เรียกใช้	
Secondary scenario: Edit device	
Administrator เลือกอุปกรณ์ที่ต้องการแก้ไขข้อมูล	
Postconditions of secondary scenario: Edit device	
หน้าเว็บแสดงผลหน้าตาบางส่วนของช่องให้กรอกข้อมูลของอุปกรณ์	
Secondary scenario: Delete device	
Administrator เลือกอุปกรณ์ที่ต้องการลบและเลือกปุ่มรูปถังขยะ	
Postconditions of secondary scenario: Delete device	
ข้อมูลอุปกรณ์ถูกลบและเรียกหน้า Device Management ใหม่	

ตารางที่ 3.23 Edit device

Use case	Edit device
ID	UC19
Actors	Administrator
Preconditions:	
<ol style="list-style-type: none"> <li>1. Administrator เข้าสู่ระบบ</li> <li>2. Administrator เรียกใช้หน้า Device Management</li> <li>3. Administrator เลือกอุปกรณ์ที่ต้องการแก้ไขข้อมูล</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. Administrator กรอกข้อมูลใหม่และกด OK <ol style="list-style-type: none"> <li>2.1 ถ้าบันทึกข้อมูลใหม่สำเร็จ เรียกหน้า Device ใหม่</li> <li>2.3 ถ้าบันทึกข้อมูลใหม่ไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Update device to database	
Postconditions:	
ข้อมูลใหม่ของอุปกรณ์ที่ HR แก้ไขถูกบันทึกในฐานข้อมูลและหน้าเว็บแสดงผลข้อมูลใหม่	
Secondary scenario: Update device to database	
ระบบ Device Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Update device to database	
ข้อมูลอุปกรณ์ในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	

## ตารางที่ 3.24 Delete device

Use case	Delete device
ID	UC20
Actors	Administrator
Preconditions:	
<ol style="list-style-type: none"> <li>1. Administrator เข้าสู่ระบบ</li> <li>2. Administrator เรียกใช้หน้า Device</li> <li>3. Administrator เลือกพนักงานที่ต้องการลบออกจากระบบ</li> </ol>	
Primary scenario:	
<ol style="list-style-type: none"> <li>1. หน้าเว็บแสดงผลส่วนของการแก้ไขข้อมูล</li> <li>2. Administrator เลือกปุ่มลบในส่วนการแก้ไขข้อมูล เพื่อลบข้อมูลอุปกรณ์ <ol style="list-style-type: none"> <li>2.1 ถ้าลบอุปกรณ์สำเร็จ เรียกหน้า Device ใหม่</li> <li>2.3 ถ้าลบอุปกรณ์ไม่สำเร็จ หน้าเว็บจะแจ้งเตือนข้อผิดพลาด</li> </ol> </li> </ol>	
Secondary scenario:	
Update device to database	
Postconditions:	
ข้อมูลของพนักงานถูกลบออกจากระบบ	
Secondary scenario: Update device to database	
ระบบ Device Management แก้ไขข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Update device to database	
ข้อมูลอุปกรณ์ในฐานข้อมูลถูกแก้ไขและส่งสถานะการแก้ไขไปให้หน้าเว็บ	

## Subsystem User Management

### ตารางที่ 3.25 Get user from database

Use case	Get user from database
ID	UC21
Actors	Employee, HR
Preconditions: Employee เรียกหน้า Personal information หรือ HR เรียกหน้า User Management	
Flow of events: <ol style="list-style-type: none"> <li>1. สร้าง Query ขึ้นให้ตรงกับข้อมูลพนักงานที่ต้องการ</li> <li>2. ฐานข้อมูลค้นหาข้อมูลพนักงานตาม Query</li> <li>3. ส่งข้อมูลพนักงานไปยังส่วนที่ขอข้อมูล</li> </ol>	
Postconditions: ข้อมูลพนักงานที่ถูกร้องขอถูกส่งไปอย่างถูกต้อง	

### ตารางที่ 3.26 Add new user to database

Use case	Add new user to database
ID	UC22
Actors	HR
Preconditions: HR เพิ่มพนักงานใหม่	
Primary scenario: <ol style="list-style-type: none"> <li>1. สร้างรหัสพนักงานให้พนักงานใหม่</li> <li>2. บันทึกข้อมูลพนักงานใหม่ลงฐานข้อมูล</li> <li>3. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน</li> </ol>	
Secondary scenario: Update location to database	
Postconditions: ข้อมูลพนักงานใหม่ถูกบันทึกลงในฐานข้อมูลพร้อมส่งสถานะการบันทึกข้อมูล	
Secondary scenario: Update location to database HR เลือกสถานที่ให้พนักงาน	
Postconditions of secondary scenario: Update location to database รหัสพนักงานถูกเพิ่มในข้อมูลสถานที่	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 3.27 Update user to database

Use case	Update user to database
ID	UC23
Actors	HR
Preconditions: HR เปลี่ยนแปลงข้อมูลพนักงาน	
Primary scenario: 1. แก้ไขหรือลบข้อมูลพนักงานในฐานข้อมูล 2. ส่งสถานะการเปลี่ยนแปลงข้อมูลไปยังส่วนที่เรียกใช้งาน	
Secondary scenario: Update location to database	
Postconditions: ข้อมูลพนักงานใหม่ถูกบันทึกลงในฐานข้อมูล หรือถูกลบออกจากฐานข้อมูล พร้อมส่งสถานะการเปลี่ยนแปลงข้อมูล	
Secondary scenario: Update location to database HR เปลี่ยนแปลงสถานที่ของผู้ใช้งาน หรือ HR ลบผู้ใช้งาน	
Postconditions of secondary scenario: Update location to database รหัสพนักงานถูกแก้ไขหรือลบจากข้อมูลสถานที่	

### ตารางที่ 3.28 Save fingerprint to database

Use case	Save fingerprint to database
ID	UC24
Actors	HR
Preconditions: HR สั่งให้เครื่องบันทึกลายนิ้วมือเก็บนิ้วมือพนักงาน	
Flow of events: 1. รอรับข้อมูลลายนิ้วมือ 2. บันทึกลายนิ้วมือไปยังข้อมูลพนักงานในฐานข้อมูล 3. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลลายนิ้วมือถูกเพิ่มไปยังข้อมูลพนักงานในฐานข้อมูล	

ตารางที่ 3.29 Clear user fingerprint from database

Use case	Clear user fingerprint from database
ID	UC25
Actors	HR
Preconditions: HR เลือก Clear fingerprint ของพนักงานคนใดคนหนึ่ง	
Flow of events: 1. ลบข้อมูลลายนิ้วมือทั้งหมดออกจากข้อมูลพนักงานที่ HR เลือก 2. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลลายนิ้วมือทั้งหมดของพนักงานที่ HR เลือกถูกลบออกจากฐานข้อมูล	

Subsystem Internal-User Management

ตารางที่ 3.30 Add new user to device

Use case	Add new user to device
ID	UC26
Actors	HR
Preconditions: HR เพิ่มพนักงานใหม่และพนักงานใหม่ถูกเลือกสถานที่	
Flow of events: 1. บันทึกข้อมูลพนักงานใหม่ลงอุปกรณ์ที่อยู่ในสถานที่ทั้งหมด 2. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลพนักงานถูกบันทึกในอุปกรณ์ทั้งหมดที่อยู่ในสถานที่ของพนักงาน	

ตารางที่ 3.31 Update user to device

Use case	Update user to device
ID	UC27
Actors	HR
Preconditions: HR เปลี่ยนแปลงข้อมูลพนักงานและพนักงานถูกเลือกสถานที่	
Flow of events: 1. แก๊ซหรือลบข้อมูลพนักงานในอุปกรณ์ที่อยู่ในสถานที่ทั้งหมด 2. ส่งสถานะการเปลี่ยนแปลงข้อมูลไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลใหม่ของพนักงานถูกเปลี่ยนแปลงในอุปกรณ์ที่อยู่ในสถานที่ทั้งหมด	

ตารางที่ 3.32 Set fingerprint to device

Use case	Set fingerprint to device
ID	UC28
Actors	HR
Preconditions: HR ย้ายสถานที่ของพนักงาน	
Flow of events: 1. เรียกข้อมูลลายนิ้วมือของพนักงานที่ HR ย้ายมาจากฐานข้อมูล 2. บันทึกลายนิ้วมือไปยังอุปกรณ์ที่อยู่ในสถานที่ใหม่	
Postconditions: ข้อมูลลายนิ้วมือของพนักงานถูกเปลี่ยนแปลงในอุปกรณ์ที่อยู่ในสถานที่ทั้งหมด	

### ตารางที่ 3.33 Set device enrolling mode

Use case	Set device enrolling mode
ID	UC29
Actors	HR
Preconditions: HR สั่งให้เครื่องบันทึกลายนิ้วมือเก็บนิ้วมือพนักงาน	
Primary scenario: 1. สั่งเครื่องบันทึกลายนิ้วมือให้อยู่ในสถานะรอรับลายนิ้วมือ 2. รอรับสถานะการเก็บลายนิ้วมือ	
Secondary scenario: Get fingerprint from device	
Postconditions: ลายนิ้วมือถูกบันทึกในอุปกรณ์	
Secondary scenario: Get fingerprint from device ระบบดึงข้อมูลลายนิ้วมือออกจากอุปกรณ์	
Postconditions of secondary scenario: Get fingerprint from device ระบบได้รับข้อมูลลายนิ้วมือจากอุปกรณ์	

### ตารางที่ 3.34 Get fingerprint from device

Use case	Get fingerprint from device
ID	UC30
Actors	HR
Preconditions: HR สั่งให้เครื่องบันทึกลายนิ้วมือเก็บนิ้วมือพนักงานและการเก็บบันทึกลายนิ้วมือสำเร็จ	
Primary scenario: 1. สั่งให้เครื่องบันทึกลายนิ้วมือส่งลายนิ้วมือของพนักงานที่ต้องการ 2. เลือกลายนิ้วมือที่ระบบต้องการ	
Secondary scenario: Save fingerprint to database	
Postconditions: ระบบได้รับข้อมูลลายนิ้วมือของพนักงานที่ต้องการจากอุปกรณ์	
Secondary scenario: Save fingerprint to database บันทึกลายนิ้วมือในข้อมูลของพนักงานในฐานข้อมูล	
Postconditions of secondary scenario: Save fingerprint to database ข้อมูลลายนิ้วมือถูกเพิ่มไปยังข้อมูลพนักงานในฐานข้อมูล	

### ตารางที่ 3.35 Clear user fingerprint from device

Use case	Clear user fingerprint from device
ID	UC31
Actors	HR
Preconditions: HR เลือก Clear fingerprint ของพนักงานคนใดคนหนึ่ง	
Flow of events: 1. ลบข้อมูลลายนิ้วมือทั้งหมดออกจากข้อมูลพนักงานในอุปกรณ์ทั้งหมดที่ HR เลือก 2. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลลายนิ้วมือทั้งหมดของพนักงานที่ HR เลือกถูกลบออกจากอุปกรณ์ทั้งหมด	

### Subsystem Attendance

ตารางที่ 3.36 Get user attendance from database

Use case	Get user attendance from database
ID	UC32
Actors	Employee, HR
Preconditions: Employee หรือ HR เรียกส่วนที่ต้องการข้อมูลบันทึกเวลาในฐานะข้อมูล	
Flow of events: <ol style="list-style-type: none"> <li>1. สร้าง Query ขึ้นให้ตรงกับข้อมูลบันทึกเวลาที่ต้องการ</li> <li>2. ฐานข้อมูลค้นหาบันทึกเวลาตาม Query</li> <li>3. ส่งข้อมูลบันทึกเวลากลับไปยังส่วนที่ขอข้อมูล</li> </ol>	
Postconditions: ข้อมูลบันทึกเวลาถูกส่งกลับไปยังส่วนที่ต้องการ	

ตารางที่ 3.37 Add attendance to database

Use case	Add attendance to database
ID	UC33
Actors	Schedule
Preconditions: Schedule สั่งให้ดึงบันทึกเวลาจากอุปกรณ์มาเก็บในฐานะข้อมูล	
Flow of events: <ol style="list-style-type: none"> <li>1. รอรับบันทึกเวลาจากอุปกรณ์</li> <li>2. สร้างรหัสบันทึกเวลาให้กับบันทึกเวลา</li> <li>3. เพิ่มบันทึกเวลาไปยังฐานข้อมูล</li> </ol>	
Postconditions: ข้อมูลบันทึกเวลาถูกบันทึกในฐานข้อมูล	

## Subsystem Device Scheduler

ตารางที่ 3.38 Set device time

Use case	Set device time
ID	UC34
Actors	Schedule
Preconditions: ถึงกำหนดเวลาที่ Schedule ถูกตั้งค่าไว้	
Flow of events: 1. ตั้งเวลาของระบบ มาเขียนเป็นคำสั่งให้อุปกรณ์ 2. สั่งให้อุปกรณ์ทุกเครื่องตั้งค่าเวลาตามเวลาที่อยู่ในคำสั่ง	
Postconditions: เวลาของทุกอุปกรณ์ถูกตั้งค่าให้ตรงกับระบบ	

ตารางที่ 3.39 Clear all attendance from device

Use case	Clear all attendance from device
ID	UC35
Actors	Schedule
Preconditions: Schedule สั่งให้ดึงบันทึกเวลาจากอุปกรณ์มาเก็บในฐานข้อมูลและบันทึกเวลาถูกบันทึกในฐานข้อมูลสำเร็จ	
Flow of events: 1. ลบบันทึกเวลาในอุปกรณ์ที่ตั้งบันทึกออกมา 2. ส่งสถานะการลบไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลบันทึกถูกลบจากอุปกรณ์	

ตารางที่ 3.40 Get attendance from device

Use case	Get attendance from device
ID	UC36
Actors	Schedule
Preconditions: Scheduler สั่งให้เก็บข้อมูลบันทึกเวลาจากอุปกรณ์ที่ทำงานอยู่ทั้งหมด	
Primary scenario: <ol style="list-style-type: none"> <li>ค้นหาเครื่องบันทึกเวลานิ้วมือที่ใช้งานอยู่</li> <li>ร้องขอบันทึกเวลาที่อยู่ในเครื่องบันทึกลายนิ้วมือ</li> <li>ส่งบันทึกเวลานิ้วมือทั้งหมดที่รวบรวมมาได้ไปยังไมโครเซอร์วิส Attendance เพื่อบันทึกลงฐานข้อมูล</li> </ol>	
Secondary scenario: <ol style="list-style-type: none"> <li>Add attendance to database</li> <li>Clear all attendance from device</li> </ol>	
Postconditions: ระบบได้รับข้อมูลบันทึกเวลาจากอุปกรณ์	
Secondary scenario: Add attendance to database บันทึกข้อมูลบันทึกเวลาลงใน Attendance database	
Postconditions of secondary scenario: Add attendance to database ข้อมูลบันทึกเวลาถูกบันทึก และส่งสถานะการบันทึกกลับไปส่วนที่เรียกใช้	
Secondary scenario: Clear all attendance from device ส่งคำสั่งล้างบันทึกเวลาในอุปกรณ์ที่ดึงข้อมูลมา	
Postconditions of secondary scenario: Clear all attendance from device อุปกรณ์ถูกล้างข้อมูลบันทึกเวลา	

## Subsystem Device Management

## ตารางที่ 3.41 Get device from database

Use case	Get device from database
ID	UC37
Actors	Administrator
Preconditions: Administrator เรียกหน้า Device information	
Flow of events: 1. สร้าง Query ขึ้นให้ตรงกับข้อมูลอุปกรณ์ที่ต้องการ 2. ฐานข้อมูลค้นหาข้อมูลอุปกรณ์ตาม Query 3. ส่งข้อมูลอุปกรณ์ไปยังส่วนที่ขอข้อมูล	
Postconditions: ข้อมูลอุปกรณ์ที่ถูกร้องขอถูกส่งไปอย่างถูกต้อง	

## ตารางที่ 3.42 Use case Add new device to database

Use case	Add new device to database
ID	UC38
Actors	Administrator
Preconditions: Administrator เพิ่มอุปกรณ์ใหม่	
Primary scenario: 1. บันทึกข้อมูลอุปกรณ์ใหม่ลงฐานข้อมูล 2. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Secondary scenario: Update location to database	
Postconditions: ข้อมูลอุปกรณ์ใหม่ถูกบันทึกลงในฐานข้อมูลพร้อมส่งสถานะการทำงานบันทึกข้อมูล	
Secondary scenario: Update location to database HR เลือกสถานที่ให้อุปกรณ์	
Postconditions of secondary scenario: Update location to database เลขซีเรียลของอุปกรณ์ถูกเพิ่มในข้อมูลสถานที่	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.43 Update device to database

Use case	Update device to database
ID	UC39
Actors	Administrator
Preconditions: Administrator เปลี่ยนแปลงข้อมูลอุปกรณ์	
Primary scenario: 1. เปลี่ยนแปลงข้อมูลอุปกรณ์ในฐานข้อมูล 2. ส่งสถานะการเปลี่ยนแปลงข้อมูลไปยังส่วนที่เรียกใช้งาน	
Secondary scenario: Update location to database	
Postconditions: ข้อมูลอุปกรณ์ใหม่ถูกบันทึกลงในฐานข้อมูล หรือถูกลบออกจากฐานข้อมูล พร้อมส่งสถานะการเปลี่ยนแปลงข้อมูล	
Secondary scenario: Update location to database Administrator เปลี่ยนแปลงสถานที่ของอุปกรณ์ หรือ Administrator ลบอุปกรณ์	
Postconditions of secondary scenario: Update location to database เลขซีเรียลของอุปกรณ์ถูกแก้ไขหรือลบจากข้อมูลสถานที่	

## Subsystem Location Management

## ตารางที่ 3.44 Get location from database

Use case	Get location from database
ID	UC40
Actors	HR
Preconditions: HR เรียกหน้า Location	
Flow of events: 1. สร้าง Query ขึ้นให้ตรงกับข้อมูลสถานที่ที่ต้องการ 2. ฐานข้อมูลค้นหาข้อมูลสถานที่ตาม Query 3. ส่งข้อมูลสถานที่ไปยังส่วนที่ขอข้อมูล	
Postconditions: ข้อมูลสถานที่ที่ถูกร้องขอถูกส่งไปอย่างถูกต้อง	

## ตารางที่ 3.45 Add new location to database

Use case	Add new location to database
ID	UC41
Actors	HR
Preconditions: HR เพิ่มสถานที่ใหม่	
Flow of events: 1. บันทึกข้อมูลสถานที่ใหม่ลงฐานข้อมูล 2. ส่งสถานะการทำงานไปยังส่วนที่เรียกใช้งาน	
Postconditions: ข้อมูลสถานที่ใหม่ถูกบันทึกลงในฐานข้อมูลพร้อมส่งสถานะการบันทึกข้อมูล	

ตารางที่ 3.46 Update location to database

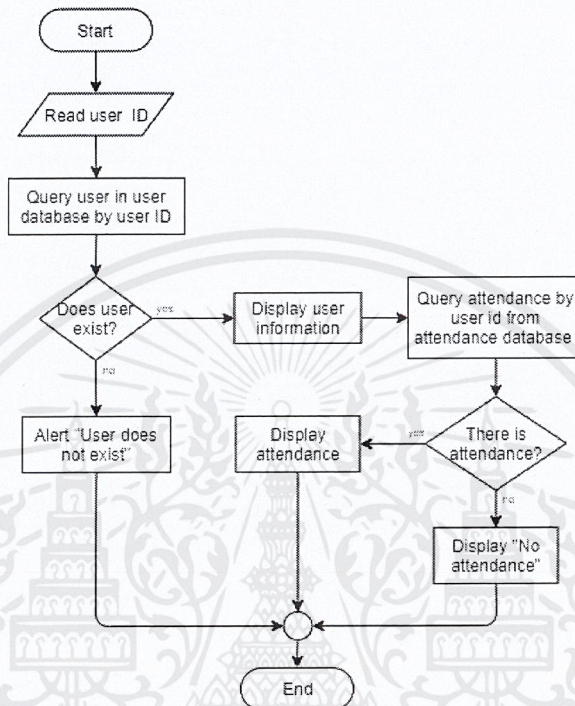
Use case	Update location to database
ID	UC42
Actors	HR
Preconditions: HR เปลี่ยนแปลงข้อมูลสถานที่	
Primary scenario: 1. เปลี่ยนแปลงข้อมูลสถานที่ในฐานข้อมูล 2. ส่งสถานะการเปลี่ยนแปลงข้อมูลไปยังส่วนที่เรียกใช้งาน	
Secondary scenario: 1. Update user to database 2. Update device to database	
Postconditions: ข้อมูลสถานที่ใหม่ถูกบันทึกลงในฐานข้อมูล หรือถูกลบออกจากฐานข้อมูล พร้อมส่งสถานะการเปลี่ยนแปลงข้อมูล	
Secondary scenario: Update user to database HR เปลี่ยนแปลงชื่อของสถานที่ ระบบเปลี่ยนชื่อสถานที่ที่อยู่ในพนักงานโดยอัตโนมัติ	
Postconditions of secondary scenario: Update user to database ชื่อสถานที่ที่อยู่ในข้อมูลพนักงานถูกเปลี่ยนอัตโนมัติ	
Secondary scenario: Update device to database HR เปลี่ยนแปลงชื่อของสถานที่ ระบบเปลี่ยนชื่อสถานที่ที่อยู่ในอุปกรณ์โดยอัตโนมัติ	
Postconditions of secondary scenario: Update device to database ชื่อสถานที่ที่อยู่ในข้อมูลอุปกรณ์ถูกเปลี่ยนอัตโนมัติ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.5.2 Flowchart

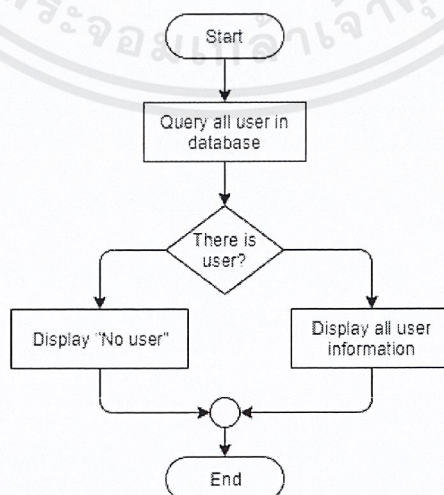
เว็บแอปพลิเคชันนั้นมีฟังก์ชันต่าง ๆ ในการจัดการข้อมูล ซึ่งแต่ละฟังก์ชันสามารถอธิบายได้ด้วยแผนภาพ Flowchart ได้ดังต่อไปนี้

## 1. ฟังก์ชัน View personal information



ภาพที่ 3.4 ฟังก์ชัน View personal information

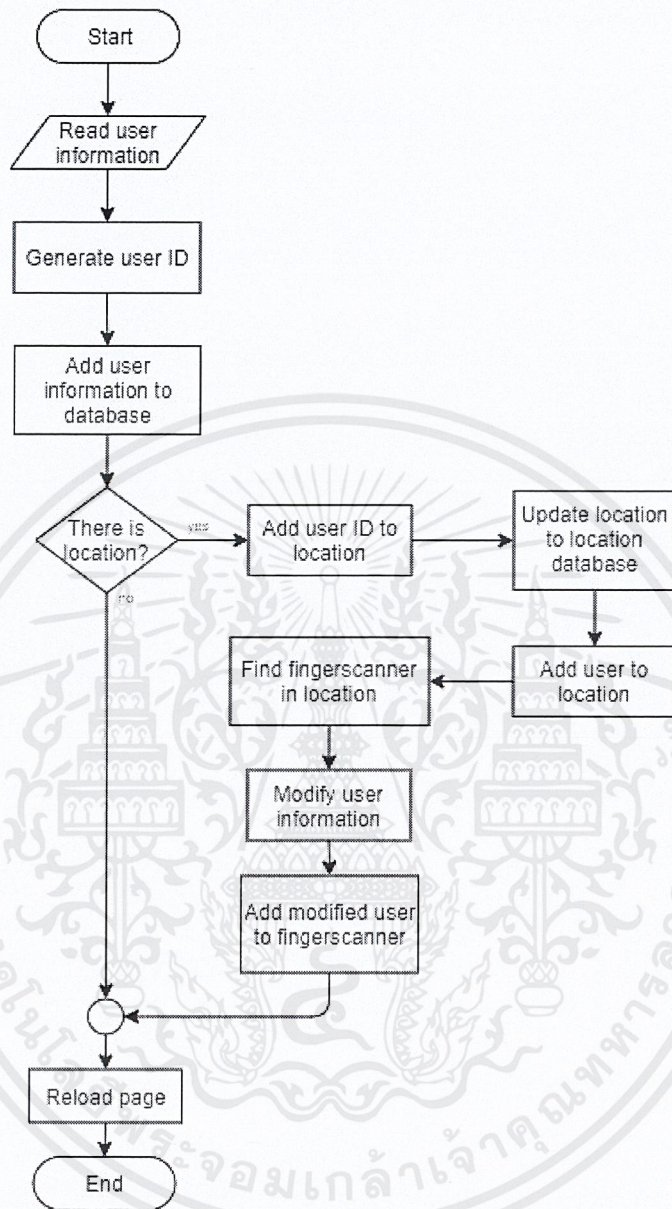
## 2. ฟังก์ชัน View user information



ภาพที่ 3.5 ฟังก์ชัน View user information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

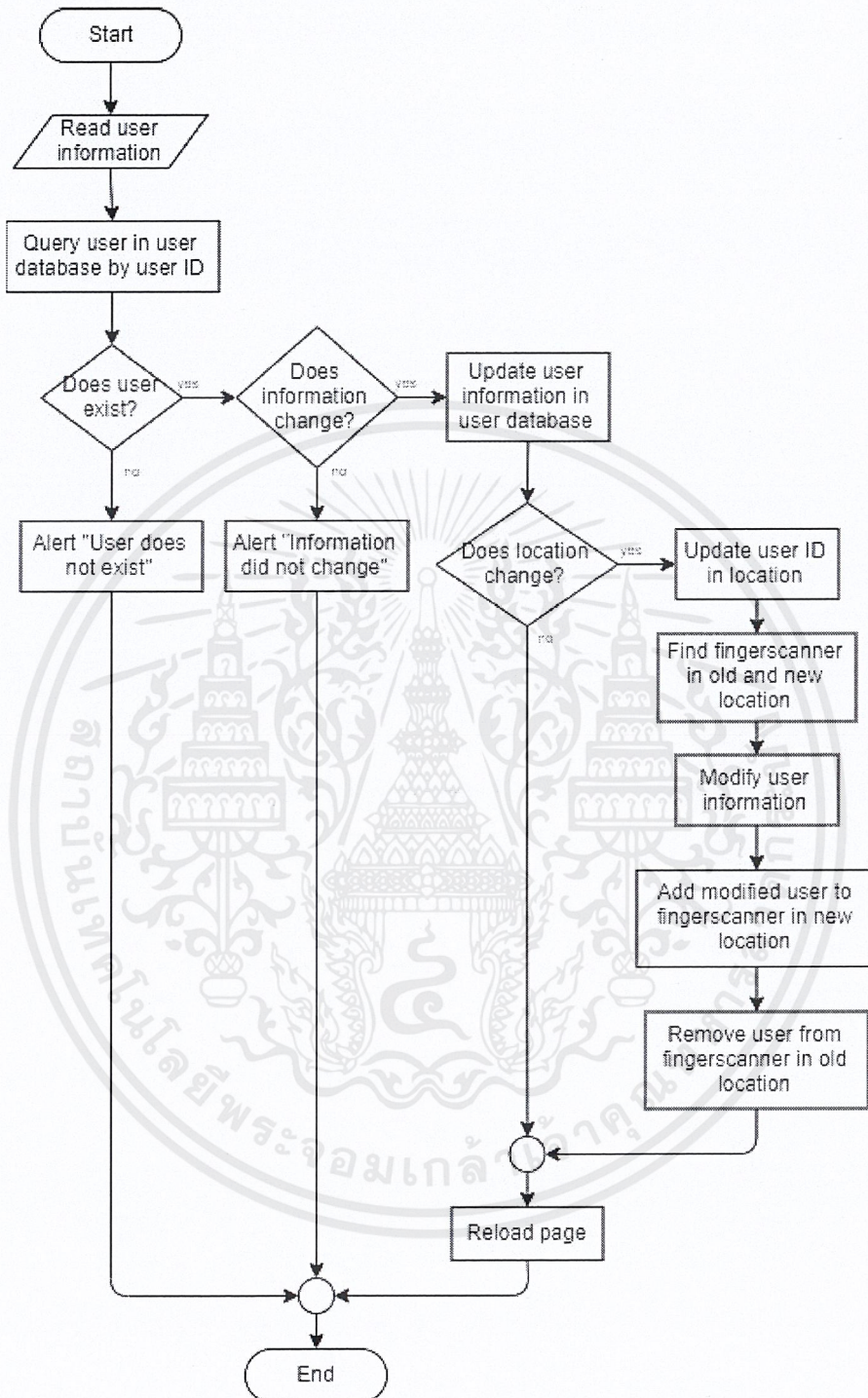
## 3. ฟังก์ชัน Create new user



ภาพที่ 3.6 ฟังก์ชัน Create new user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

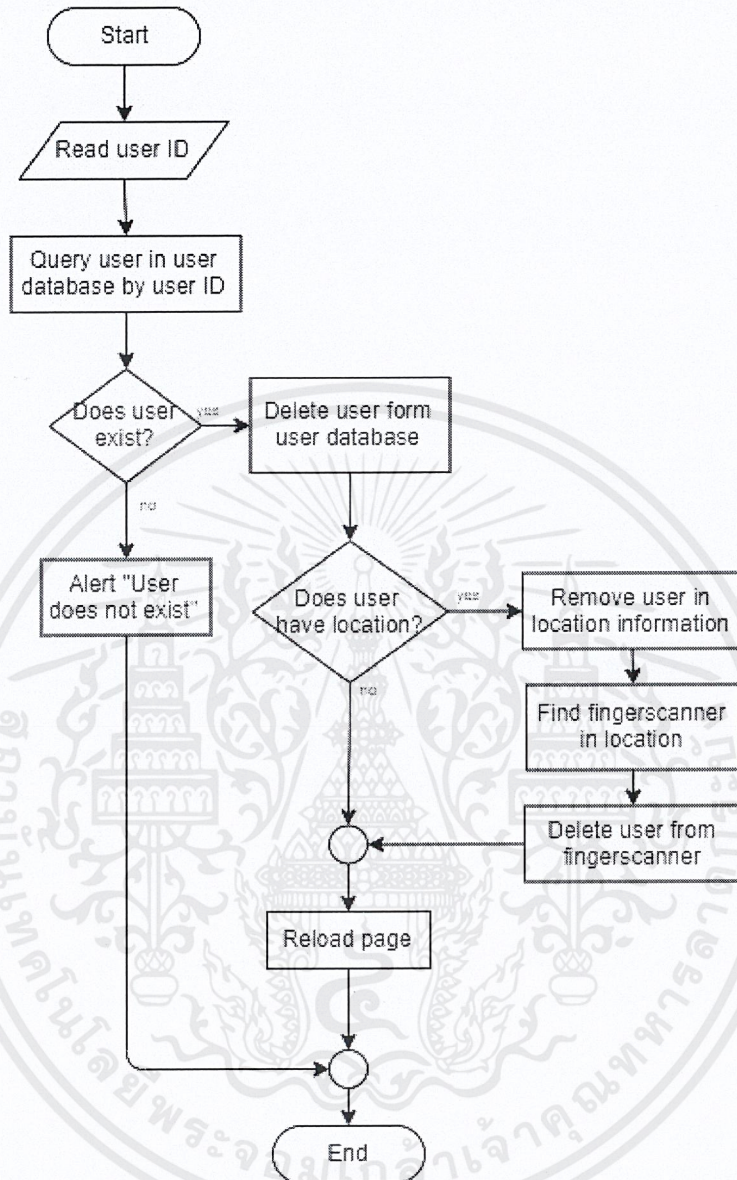
## 4. ฟังก์ชัน Edit user



ภาพที่ 3.7 ฟังก์ชัน Edit user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

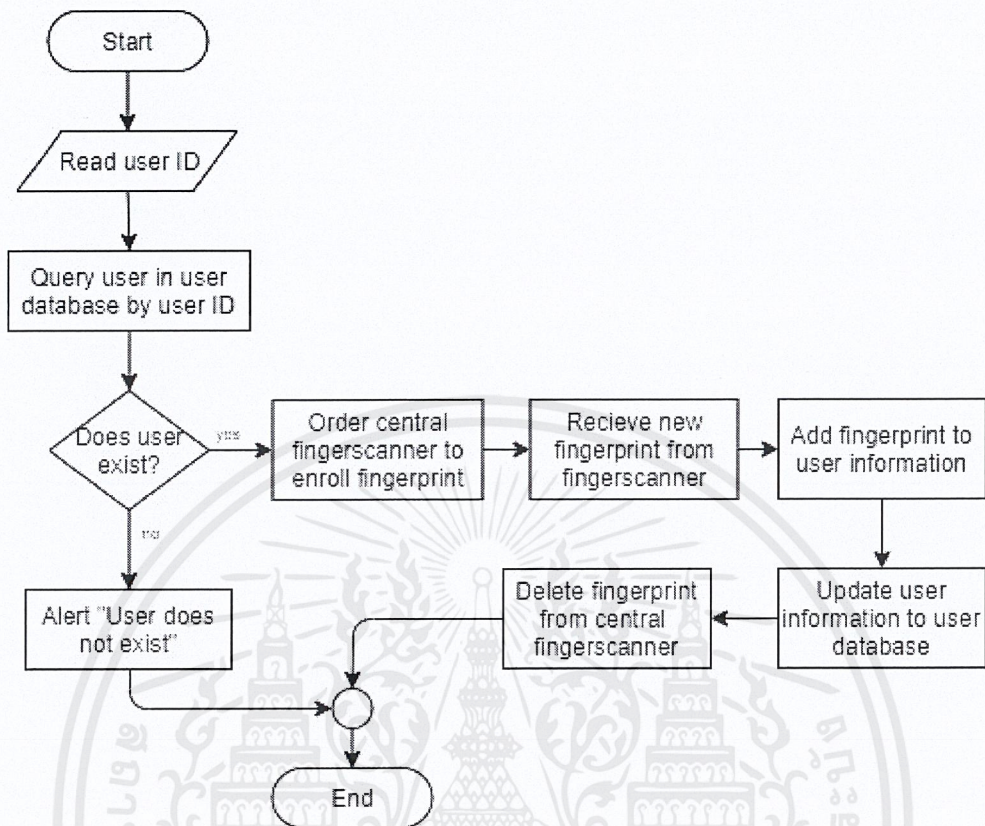
## 5. ฟังก์ชัน Delete user



ภาพที่ 3.8 ฟังก์ชัน Delete user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

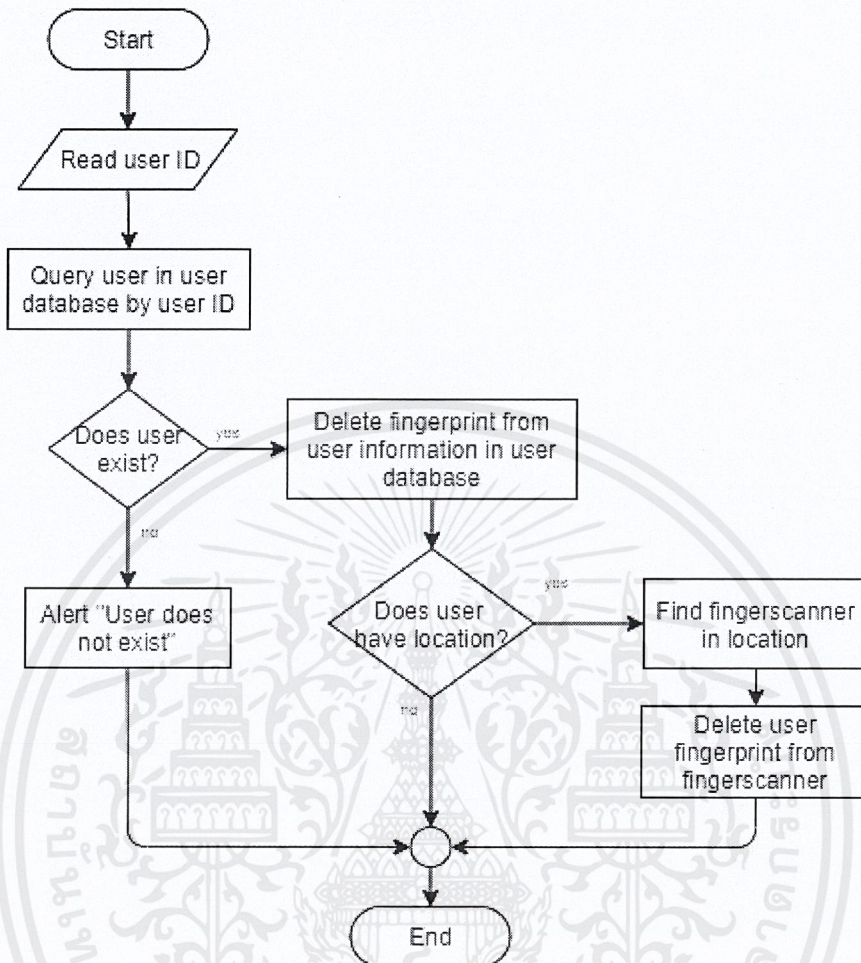
## 6. ฟังก์ชัน Enroll fingerprint



ภาพที่ 3.9 ฟังก์ชัน Enroll fingerprint

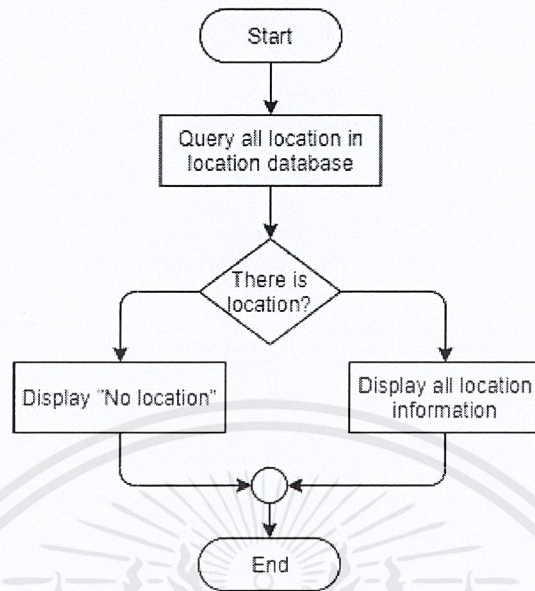
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7. ฟังก์ชัน Clear fingerprint



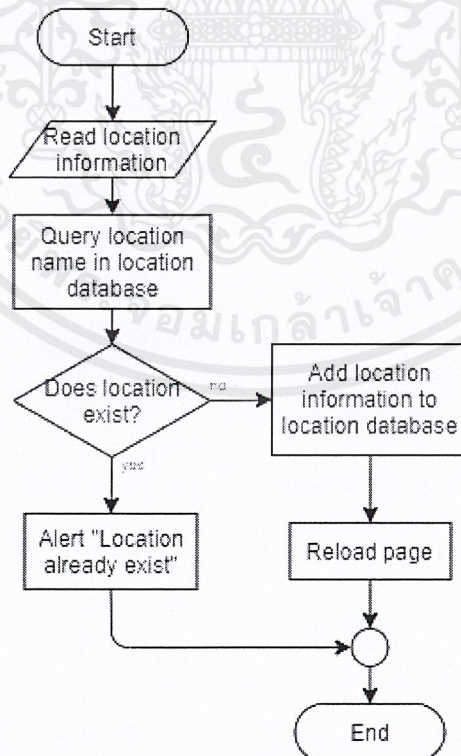
ภาพที่ 3.10 ฟังก์ชัน Clear fingerprint

## 8. ฟังก์ชัน View Location



ภาพที่ 3.11 ฟังก์ชัน View Location

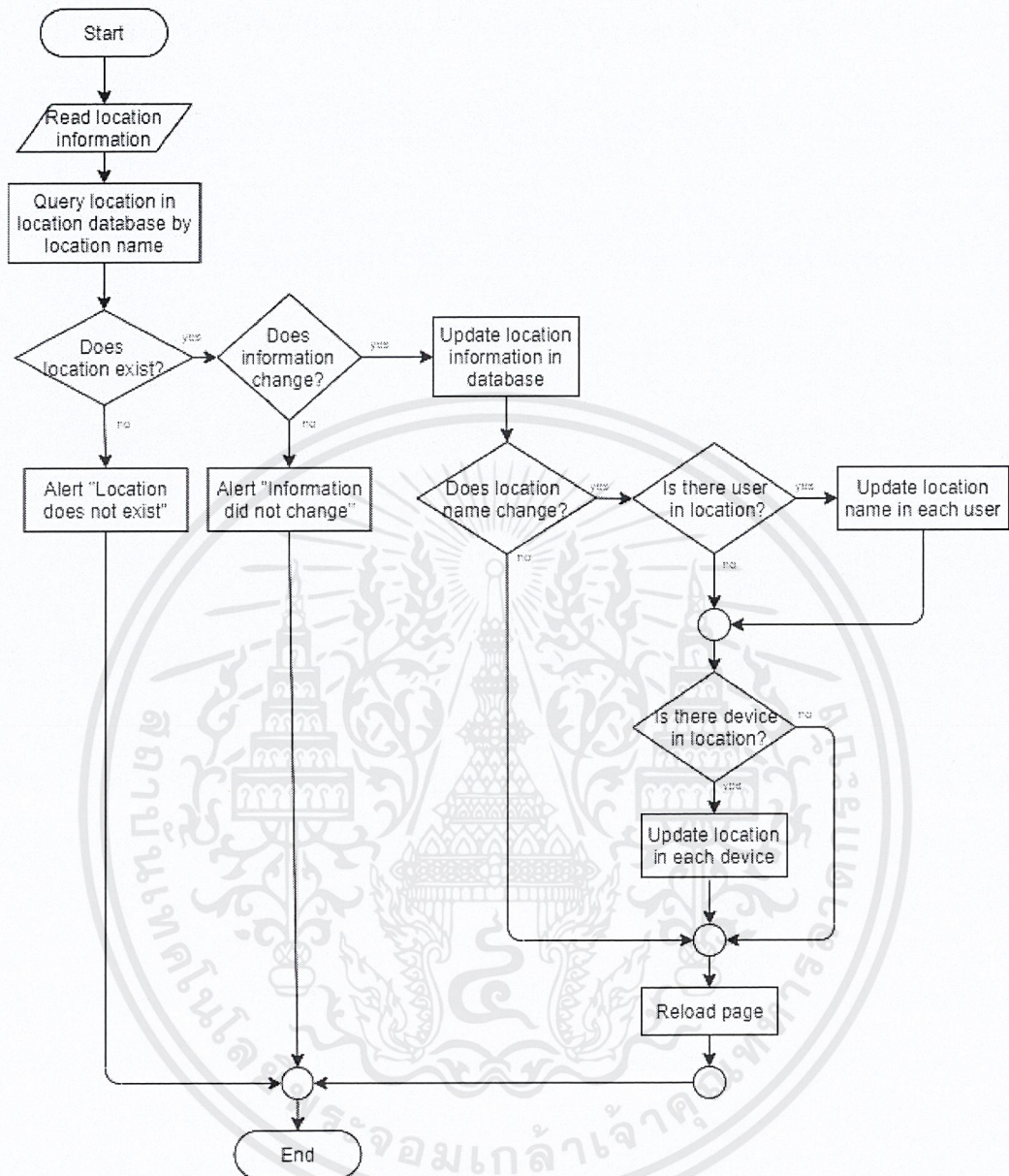
## 9. ฟังก์ชัน Create new Location



ภาพที่ 3.12 ฟังก์ชัน Create new Location

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

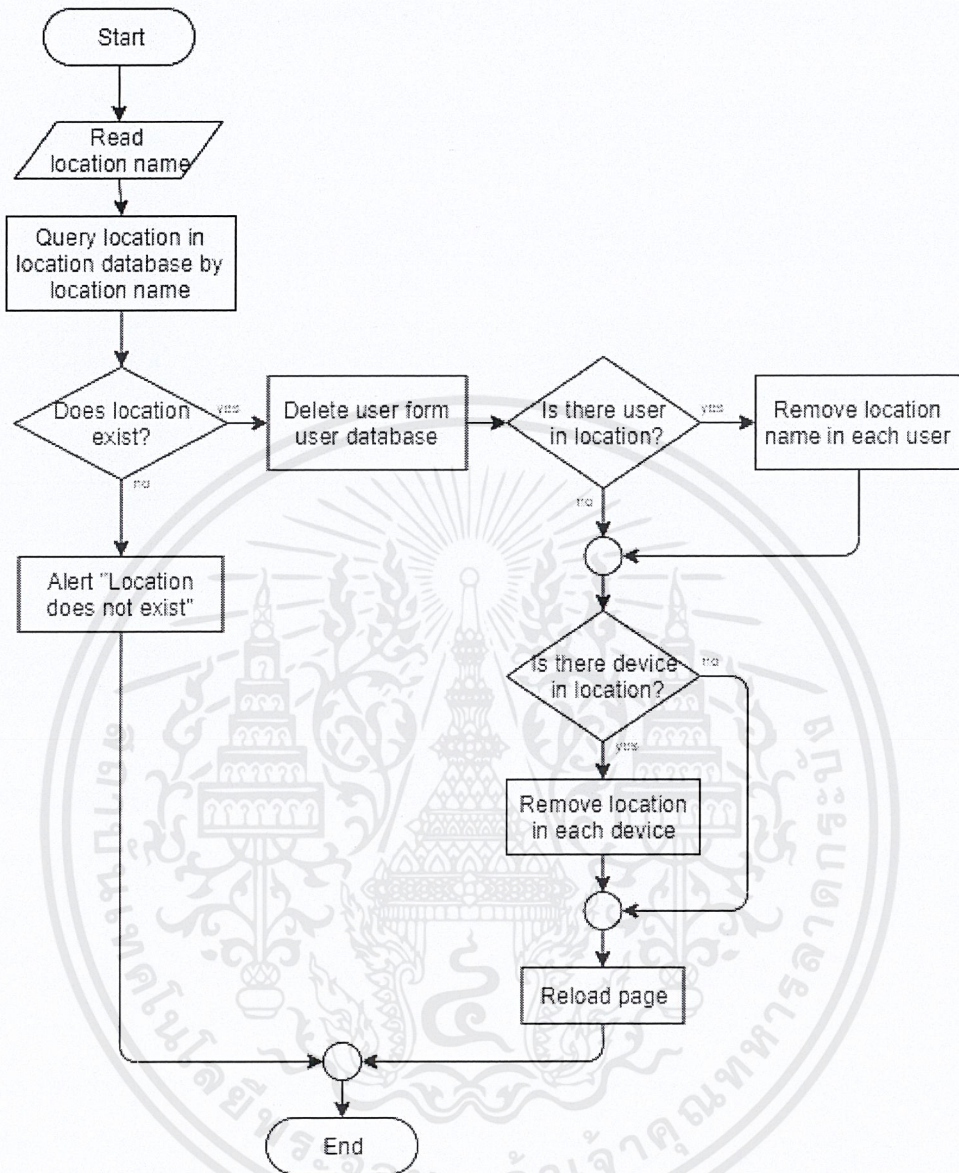
## 10. ฟังก์ชัน Edit Location



ภาพที่ 3.13 ฟังก์ชัน Edit Location

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

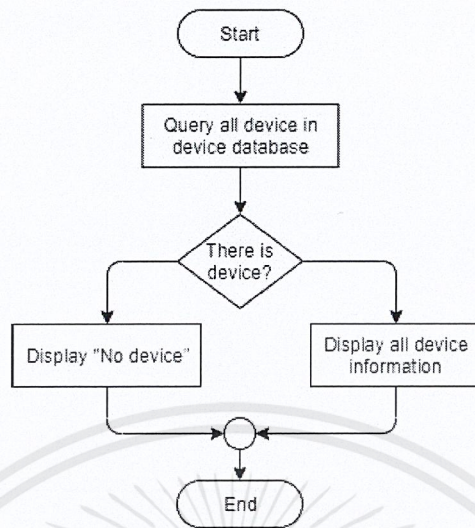
## 11. ฟังก์ชัน Delete Location



ภาพที่ 3.14 ฟังก์ชัน Delete Location

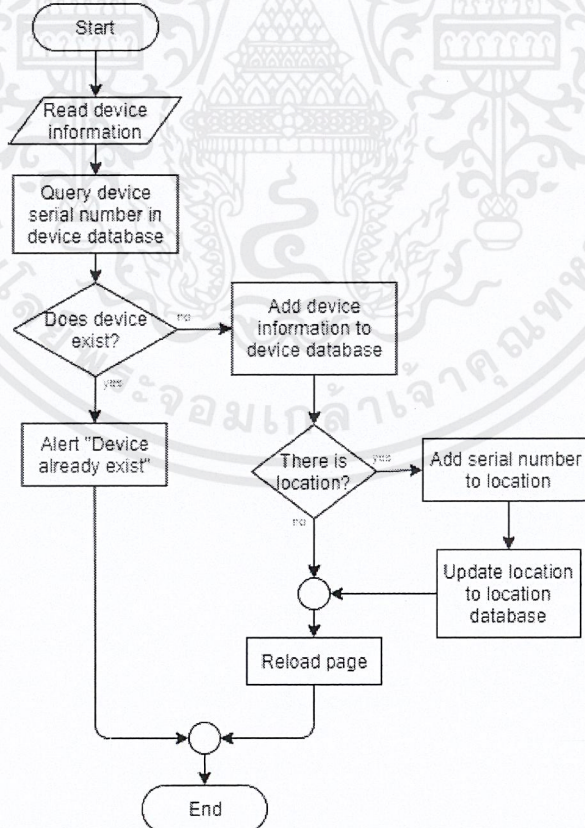
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 12. ฟังก์ชัน View Device



ภาพที่ 3.15 ฟังก์ชัน View Device

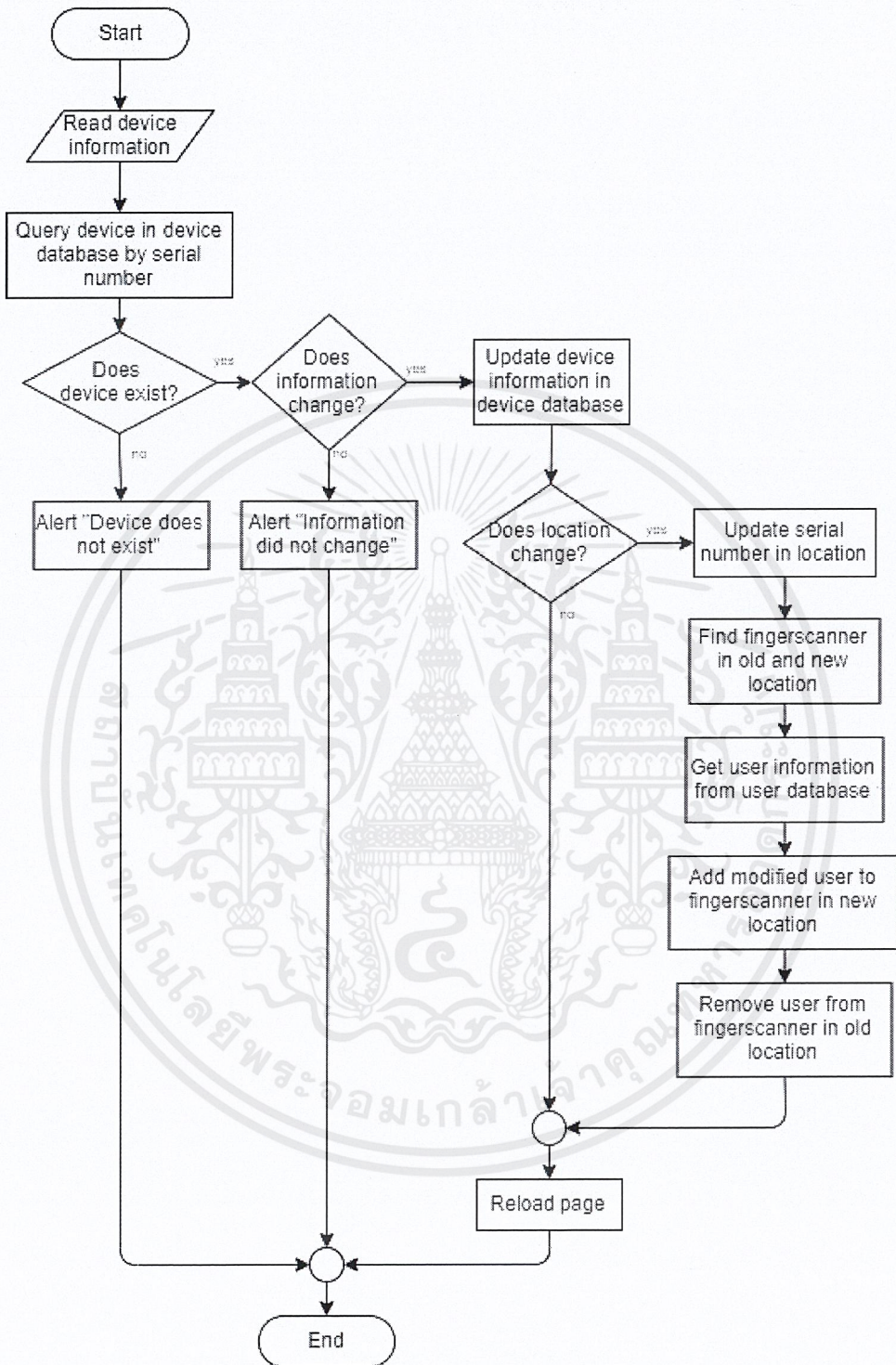
## 13. ฟังก์ชัน Create new Device



ภาพที่ 3.16 ฟังก์ชัน Create new Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

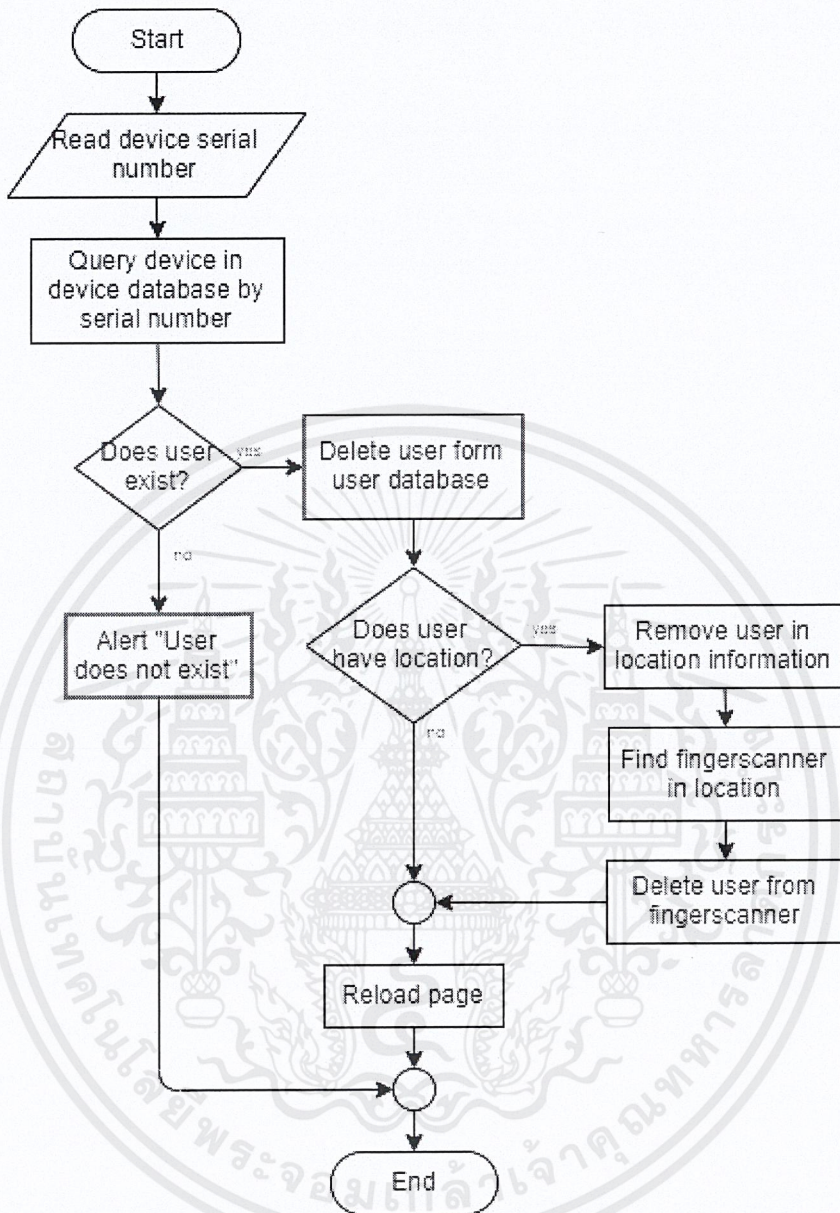
## 14. ฟังก์ชัน Edit Device



ภาพที่ 3.17 ฟังก์ชัน Edit Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

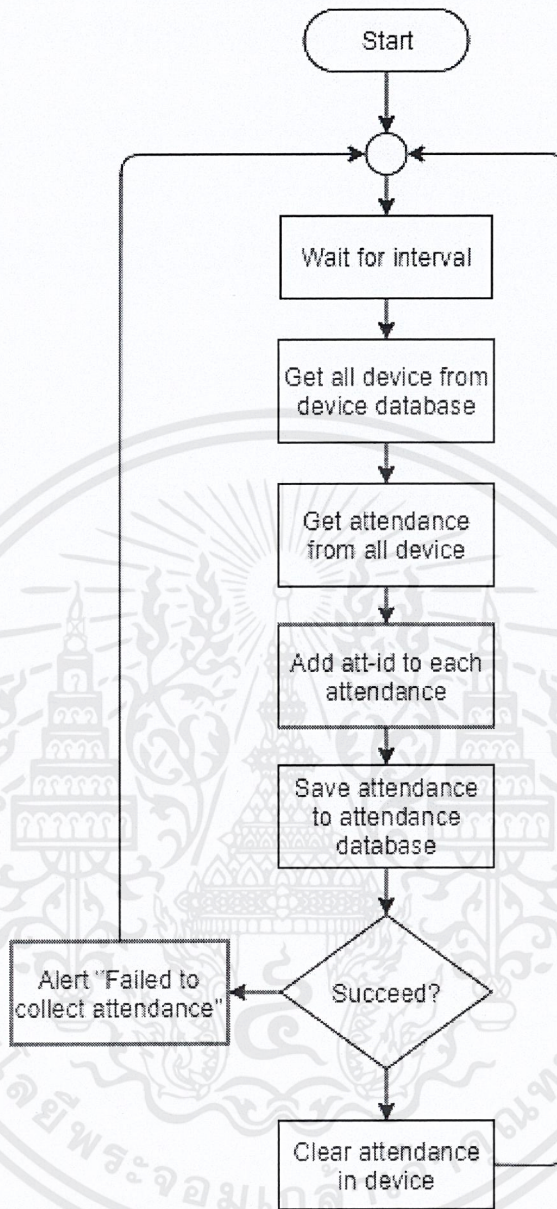
## 15. ฟังก์ชัน Delete Device



ภาพที่ 3.18 ฟังก์ชัน Delete Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 16. ฟังก์ชัน Collecting Attendance

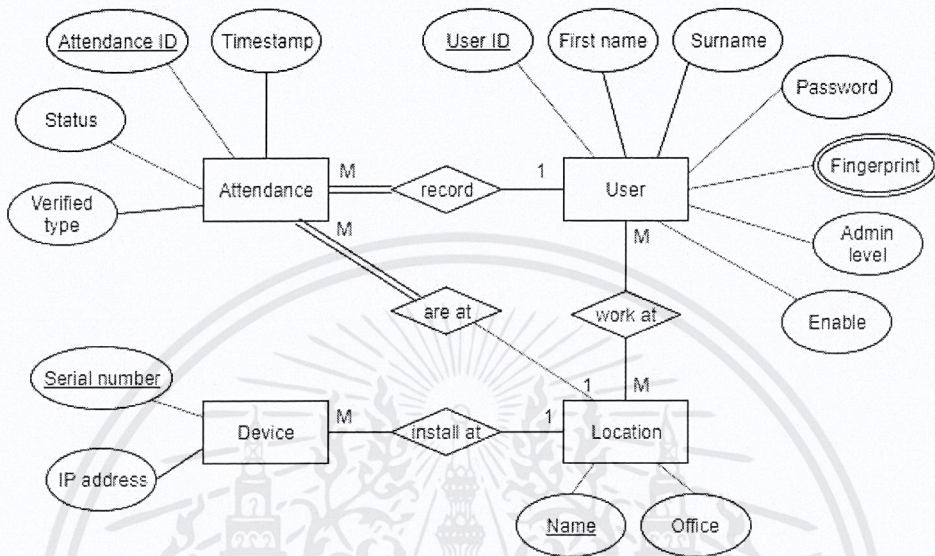


ภาพที่ 3.19 ฟังก์ชัน Collecting Attendance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3 Entity Relationship Diagrams

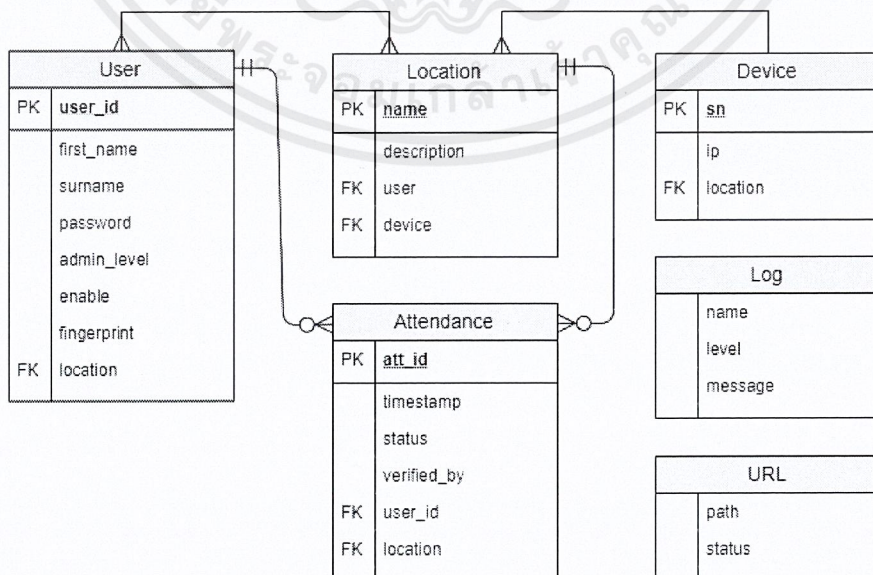
ในส่วนของการออกแบบฐานข้อมูล ผู้วิจัยได้ใช้แผนภาพความสัมพันธ์ของเอนทิตีมาใช้อธิบายโครงสร้างของฐานข้อมูล เพื่อให้เห็นความสัมพันธ์ของข้อมูลต่าง ๆ ได้อย่างชัดเจนและเข้าใจง่าย โดยมีรายละเอียดดังภาพที่ 3.20



ภาพที่ 3.20 ความสัมพันธ์ระหว่างเอนทิตีของระบบการจัดการบันทึกเวลาเข้า - ออกงาน

### 3.5.4 Database Schema Diagram

ผู้วิจัยได้นำแผนภาพ Schema มาใช้อธิบายการออกแบบและโครงสร้างของฐานข้อมูลทั้งหมด โดยมีการใช้ฐานข้อมูลและองค์ประกอบดังภาพที่ 3.21



ภาพที่ 3.21 แผนภาพ Database Schema ของระบบการจัดการบันทึกเวลาเข้า - ออกงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 3.21 แสดงถึงโครงสร้างฐานข้อมูลทั้งหมดของเว็บแอปพลิเคชัน ซึ่งแต่ละตารางนั้นมีรายละเอียดดังนี้

1. ตาราง User ใช้สำหรับเก็บข้อมูลส่วนตัว ค่าสถานะต่าง ๆ และลายนิ้วมือของพนักงานทั้งหมด

ตารางที่ 3.47 ตาราง User

Field	Key	Datatype	Description
user_id	PK	string	รหัสพนักงาน
first_name		string	ชื่อพนักงาน
surname		string	นามสกุลพนักงาน
password		string	รหัสผ่าน
admin_level		integer	ค่าความเป็นผู้ดูแลระบบ
enable		bool	สถานะการใช้งาน
fingerprint		list: string	ลายนิ้วมือ
location	FK	list: string	ชื่อสถานที่ที่พนักงานทำงาน

2. ตาราง Attendance ใช้สำหรับเก็บบันทึกเวลาทั้งหมด

ตารางที่ 3.48 ตาราง Attendance

Field	Key	Datatype	Description
att_id	PK	integer	รหัสบันทึกเวลา
timestamp		date	วันและเวลาที่พนักงานลงเวลา
status		string	สถานะการลงเวลา
verified_by		string	ประเภทที่การลงเวลา
user_id	FK	string	รหัสพนักงาน
location	FK	string	ชื่อสถานที่ที่พนักงานลงเวลา

## 3. ตาราง Location ใช้สำหรับเก็บข้อมูลสถานที่ทั้งหมด

ตารางที่ 3.49 ตาราง Location

Field	Key	Datatype	Description
name	PK	string	ชื่อสถานที่
description		string	รายละเอียดที่อยู่ของสถานที่
user		list: string	พนักงานที่มีสถานที่นี้
device		list: string	อุปกรณ์ที่มีสถานที่นี้

## 4. ตาราง Location ใช้สำหรับเก็บข้อมูลสถานที่ทั้งหมด

ตารางที่ 3.50 ตาราง Device

Field	Key	Datatype	Description
sn	PK	string	เลขซีเรียลของอุปกรณ์
ip		string	ไอพีแอดเดรสของอุปกรณ์
location		string	สถานที่ของอุปกรณ์

## 5. ตาราง Log ใช้สำหรับเก็บข้อมูลสถานที่ทั้งหมด

ตารางที่ 3.51 ตาราง Log

Field	Key	Datatype	Description
name		string	ชื่อฟังก์ชันที่เขียน Log
level		string	ระดับของ Log
message		string	คำอธิบาย

## 6. ตาราง URL ใช้สำหรับเก็บข้อมูลสถานที่ทั้งหมด

ตารางที่ 3.52 ตาราง URL

Field	Key	Datatype	Description
path		string	URL พื้นฐาน
status		string	สถานะการเรียกใช้

### 3.6 การติดต่อสื่อสารกับเครื่องบันทึกถายนิ้วมือ

เนื่องจากเครื่องบันทึกถายนิ้วมือที่ใช้ในระบบนี้เป็นอุปกรณ์ Embedded Systems สมัยเก่า มีฐานข้อมูลในตัวของอุปกรณ์ มีโปรโตคอลการสื่อสารกับซอฟต์แวร์ ซึ่งอุปกรณ์นี้มีไลบรารีในการช่วยจัดการในส่วนของการเขียนโปรแกรมเครือข่ายให้ แต่ไลบรารีดังกล่าวรองรับการพัฒนาซอฟต์แวร์ด้วย .NET Framework และมีการตอบสนองช้า ส่งผลให้ประสิทธิภาพของระบบไม่ดีเท่าที่ควร ผู้วิจัยจึงได้รับมอบหมายให้ลอกเลียนแบบการสื่อสาร โดยทำการตรวจจับ Packet การสื่อสารในเครือข่ายด้วยโปรแกรม Wireshark เพื่อวิเคราะห์โปรโตคอลการสื่อสารเพื่อสร้างเลียนแบบและสื่อสารเพื่อใช้งานฟังก์ชันที่อยู่ในอุปกรณ์ได้

การสื่อสารกับอุปกรณ์นี้ใช้ UDP ในการรับ - ส่งข้อมูลกันโดยการสื่อสารถูกออกแบบโดยใช้สถาปัตยกรรมแบบไคลเอนต์ - เซิร์ฟเวอร์ โดยเครื่องบันทึกถายนิ้วมือเป็นฝั่งเซิร์ฟเวอร์และฝั่งที่เรียกใช้งานฟังก์ชันบนอุปกรณ์เป็นฝั่งไคลเอนต์ โดยทางผู้วิจัยได้จับคู่มือโปรโตคอลการสื่อสารซึ่งบอกชุดคำสั่งที่สามารถส่งผ่านเครือข่ายเพื่อให้อุปกรณ์เข้าใจว่าผู้วิจัยต้องการใช้ฟังก์ชันใด หรือต้องการข้อมูลอะไรในฐานข้อมูลของอุปกรณ์ ซึ่งชุดคำสั่งนี้จะระบุในส่วนของข้อมูล (Payload) ใน Packet รวมถึงค่าพารามิเตอร์ของฟังก์ชันต่าง ๆ ที่ผู้วิจัยนั้นต้องวิเคราะห์รูปแบบเพื่อสามารถเลียนแบบการสื่อสารได้อย่างถูกต้อง

วิธีการดักจับ Packet ข้อมูลที่อยู่ในเครือข่ายทำได้โดยการเขียนโปรแกรมด้วยการใช้ไลบรารีที่ใช้กับ .NET Framework ซึ่งผู้วิจัยได้ใช้ภาษา C# ในการเขียนโปรแกรมเรียกใช้ฟังก์ชันต่าง ๆ และใช้โปรแกรม Wireshark ในการดักจับ Packet เพื่อนำมาวิเคราะห์โครงสร้างของ Packet

ชุดคำสั่งที่อยู่ในส่วนของข้อมูลใน Packet นั้น มีโครงสร้างดังนี้

Start	Payload size	Zero (null bytes)	Payload				
			Command ID	Checksum	Session ID	Reply number	Data / Parameter
0x5D50827d	0xYYYY	0x0000					
4 bytes	2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	(payload size - 8) bytes

ภาพที่ 3.22 โครงสร้าง Packet ที่ใช้ในการสื่อสารกับเครื่องบันทึกถายนิ้วมือ

ในส่วนของชุดคำสั่งนั้น ผู้วิจัยได้ศึกษาจากในคู่มือและเลือกคำสั่งที่จำเป็นต่อการวิจัย  
ได้ดังต่อไปนี้

ตารางที่ 3.53 ชุดคำสั่งที่ใช้ระบบ

Command	คำอธิบาย	ค่าคำสั่ง (ฐาน 10)	ค่าคำสั่ง (ฐาน 16)
CMD_CONNECT	เปิดการเชื่อมต่อ	1000	0x03e8
CMD_EXIT	ตัดการเชื่อมต่อ	1001	0x03e9
CMD_ENABLEDEVICE	เปลี่ยนสถานะเป็นการทำงานปกติ	1002	0x03ea
CMD_DISABLEDEVICE	หยุดรับข้อมูลทางลายนิ้วมือและแป้น	1003	0x03eb
CMD_REFRESHDATA	รีเฟรชข้อมูลภายในอุปกรณ์	1013	0x03f5
CMD_PREPARE_DATA	เตรียมตัวสำหรับการขนส่งข้อมูล	1500	0x05dc
CMD_DATA	Packet ข้อมูล	1501	0x05dd
CMD_FREE_DATA	ล้างบัฟเฟอร์ที่ใช้เตรียมขนส่งข้อมูล	1502	0x05de
CMD_DATA_WRRQ	อ่าน/เขียนข้อมูลขนาดใหญ่	1503	0x05df
CMD_DATA_RDY	บ่งบอกว่าพร้อมที่จะรับข้อมูลแล้ว	1504	0x05e0
CMD_ACK_OK	การร้องขอสำเร็จ	2000	0x07d0
CMD_ACK_ERROR	การร้องขอเกิดข้อผิดพลาด	2001	0x07d1
CMD_USER_WRQ	เพิ่มข้อมูลผู้ใช้งาน	8	0x0008
CMD_USERTEMP_RRQ	อ่านลายนิ้วมือของผู้ใช้งาน	9	0x0009
CMD_USERTEMP_WRQ	เพิ่มลายนิ้วมือผู้ใช้งาน	10	0x000a
CMD_ATTLOG_RRQ	ร้องขอบันทึกเวลา	11	0x000b
CMD_CLEAR_ATTLOG	ล้างบันทึกเวลาทั้งหมด	13	0x000d
CMD_DELETE_USER	ลบผู้ใช้งาน	18	0x0012
CMD_DELETE_USERTEMP	ลบลายนิ้วมือของผู้ใช้งาน	19	0x0013
CMD_CLEAR_ADMIN	ล้างสถานะของผู้ดูแล	20	0x0014
CMD_STARTENROLL	รอรับการลงทะเบียนนิ้วใหม่	61	0x003d
CMD_CANCELCAPTURE	ยกเลิกการลงทะเบียนนิ้ว	62	0x003e
CMD_TMP_WRITE	ย้ายข้อมูลลายนิ้วมือจากบัฟเฟอร์	87	0x0057
CMD_SET_TIME	ตั้งค่าเวลาให้กับอุปกรณ์	202	0x00ca

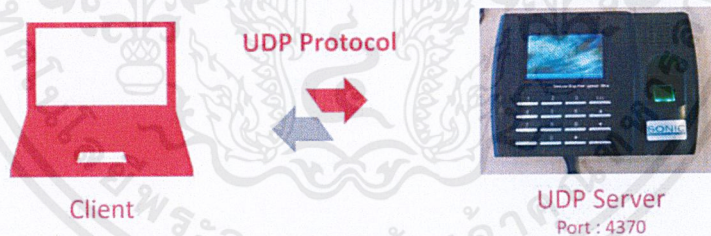
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนที่เหลือของ Payload นั้นคือส่วนของการ Checksum ในภาพที่ 3.22 นั้นใช้หลักการเดียวกันกับการ Checksum ทั่วไป ส่วนของ Session ID จะใส่ค่าเป็น 0x0000 ไปเมื่อเชื่อมต่อครั้งแรก เซิร์ฟเวอร์จะส่ง Session ID มาให้และต้องใช้ค่านี้ไปจนกว่าจะปิดการเชื่อมต่อ โดยแต่ละการเชื่อมต่อจะได้ Session ID ไม่เหมือนกัน และอุปกรณ์สามารถต่อได้เพียงหนึ่งเครื่องในเวลาเดียวกันเท่านั้น ส่วนของ Reply number จะเป็นการนับเลขการตอบกลับของการสื่อสาร ใช้สำหรับอ้างอิงว่า Packet นี้เป็นการตอบกลับการร้องขอหรือคำสั่งใด

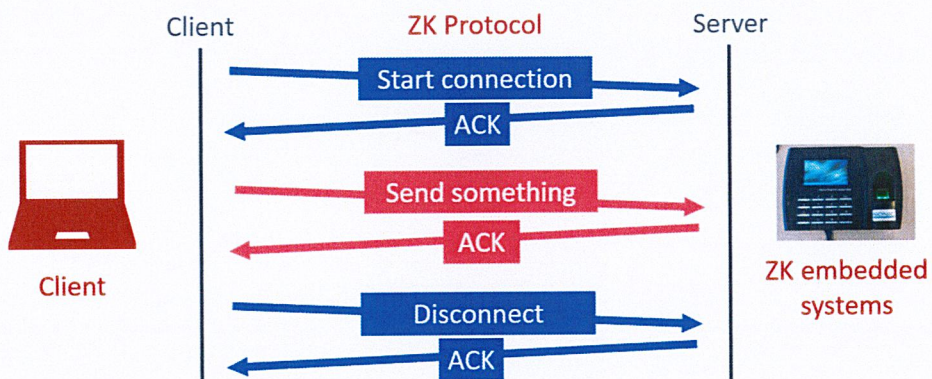
ส่วนสุดท้ายส่วน Data หรือ Parameter นั้น ชุดคำสั่งหลายคำสั่งต้องการส่งพารามิเตอร์หรือข้อมูลไปเก็บในอุปกรณ์ เช่น การเพิ่มพนักงาน ซึ่งผู้วิจัยต้องวิเคราะห์ว่าแต่ละคำสั่งนั้นต้องใส่พารามิเตอร์อะไรบ้างและมีรูปแบบของการใส่พารามิเตอร์อย่างไรที่ทำให้อุปกรณ์เข้าใจคำสั่ง

การเชื่อมต่อเพื่อสั่งการให้เครื่องบันทึกถายนิ้วมือกระทำในฟังก์ชันที่ต้องการนั้นควรอยู่ภายใต้สถานะป้องกันการรับข้อมูลเข้าจากทางอื่น เช่น การลงเวลางานโดยพิมพ์ลายนิ้วมือ ซึ่งจะส่งผลให้การทำงานนั้นมีผลลัพธ์ที่ผิดเพี้ยนออกไป ซึ่งผู้วิจัยได้สรุปไว้ว่า เมื่อมีโคลเอนต์เชื่อมต่อกับเครื่องบันทึกถายนิ้วมือ จะสั่งให้อุปกรณ์จะปิดส่วนการรับข้อมูลจนกว่าจะตัดการเชื่อมต่อ

เนื่องจากอุปกรณ์นั้นมีข้อจำกัดด้านเนื้อที่หน่วยความจำ ซึ่งระบบการจัดการบันทึกเวลาการเข้า – ออกงานนี้ได้ออกแบบโดยการนำข้อมูลที่จำเป็นมาใส่ในเครื่องบันทึกถายนิ้วมือ เพื่อลดการใช้ทรัพยากรที่ไม่จำเป็นและเหตุผลทางด้านอื่น ๆ อีกด้วย



ภาพที่ 3.23 โพรโตคอล UDP ของการสื่อสารระหว่างอุปกรณ์กับผู้ที่ใช้



ภาพที่ 3.24 การสื่อสารระหว่างอุปกรณ์กับผู้ที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4 ผลการวิจัย

### 4.1 เว็บไซต์แอปพลิเคชัน

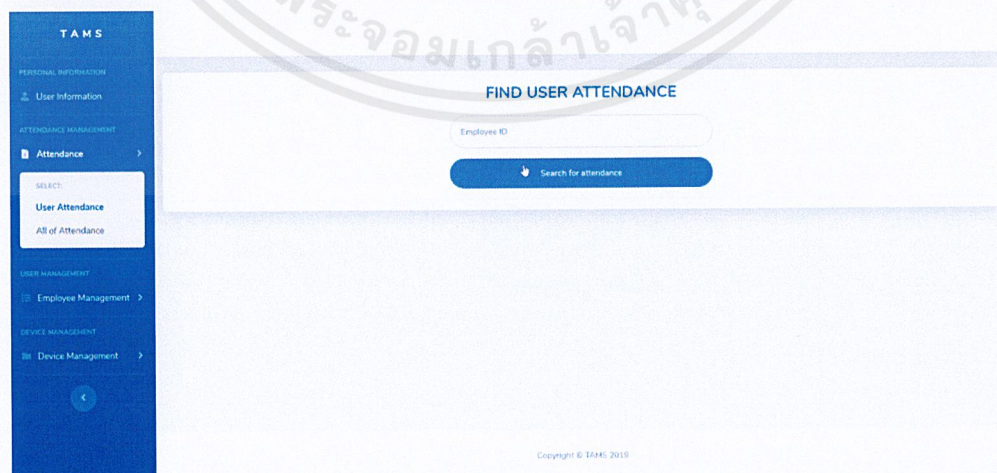
ผู้วิจัยได้พัฒนาเว็บไซต์แอปพลิเคชันของระบบในส่วนของการติดต่อกับผู้ใช้งานของเว็บไซต์แอปพลิเคชันนั้น โดยใช้หน้าเว็บไซต์สำเร็จรูป ผู้วิจัยได้พัฒนาเว็บไซต์แอปพลิเคชันให้ใช้งานในส่วนจำเป็นเพื่อทดสอบการทำงานดังนี้

1. ส่วนของการเรียกดูข้อมูลส่วนตัว (Personal Information)



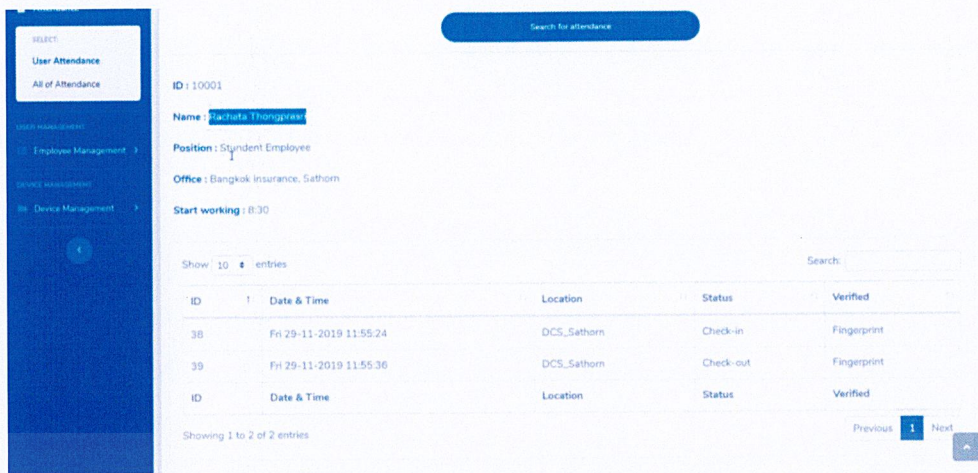
ภาพที่ 4.1 จอภาพหน้า Personal Information (Index)

2. ส่วนของการเรียกดูข้อมูลบันทึกเวลาของพนักงาน (Attendance)



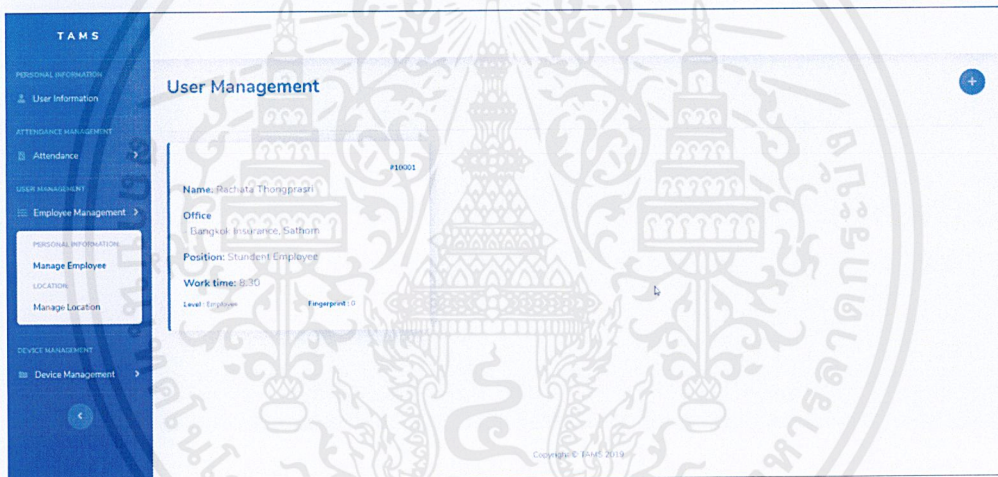
ภาพที่ 4.2 จอภาพหน้า User Attendance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

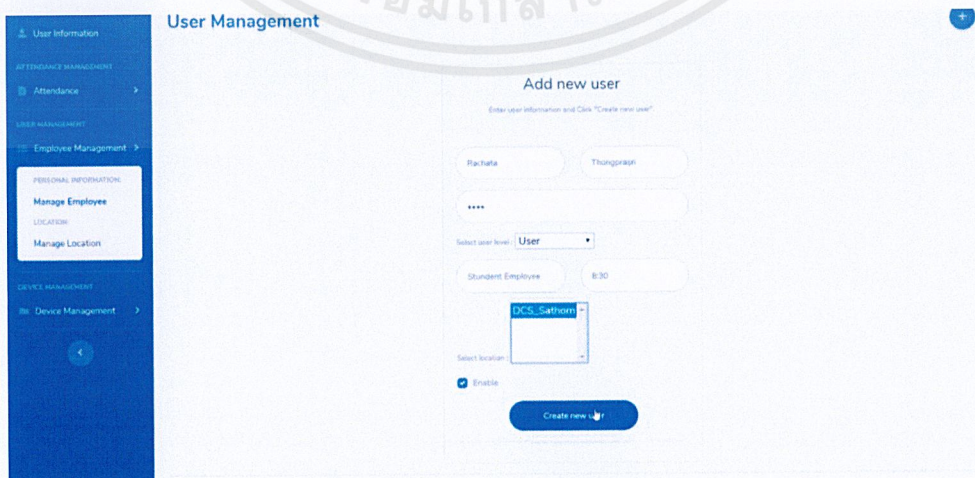


ภาพที่ 4.3 จอภาพการเรียกดูข้อมูลบันทึกเวลาของพนักงานหน้า User Attendance

### 3. ส่วนของการจัดการพนักงาน (User Management)

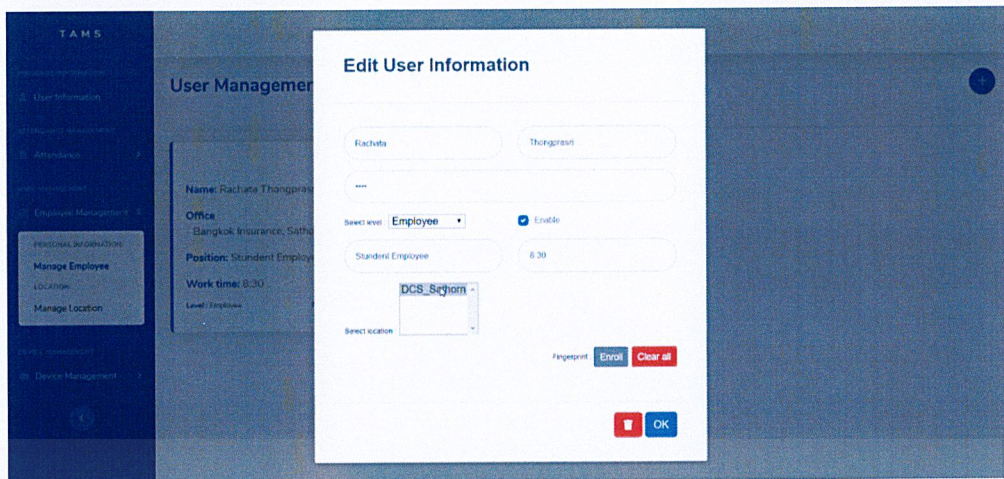


ภาพที่ 4.4 จอภาพหน้า User Management



ภาพที่ 4.5 จอภาพการเพิ่มพนักงานในหน้า User Management

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

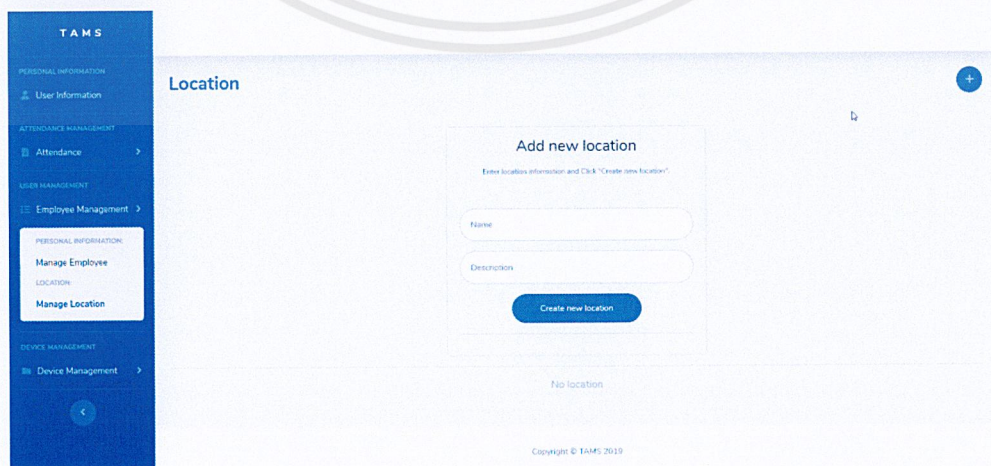


ภาพที่ 4.6 จอภาพการแก้ไขข้อมูลพนักงานในหน้า User Management

### 3. ส่วนของการจัดการสถานที่ (Location)



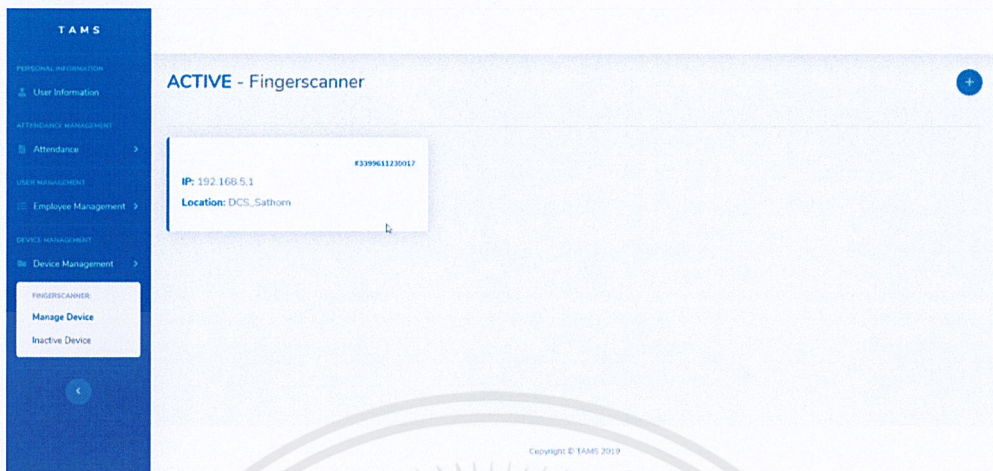
ภาพที่ 4.7 จอภาพหน้า Location



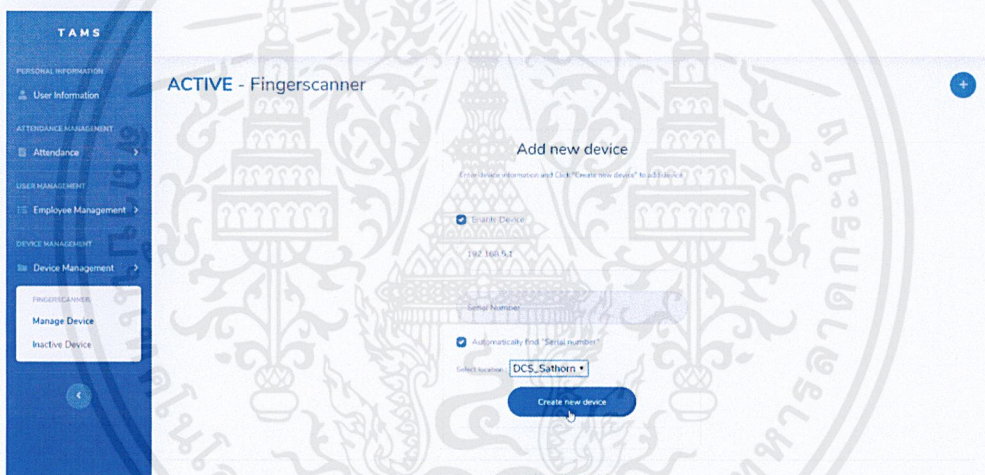
ภาพที่ 4.8 จอภาพการเพิ่มสถานที่ในหน้า Location

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4. ส่วนของการจัดการอุปกรณ์ (Device Management)



ภาพที่ 4.9 จอภาพหน้า Device



ภาพที่ 4.10 จอภาพการเพิ่มอุปกรณ์ในหน้า Device

การใช้งานเว็บแอปพลิเคชันนี้สามารถใช้งานฟังก์ชันต่าง ๆ ในระบบเมื่อเทียบกับความต้องการที่รวบรวมมาอยู่ที่ประมาณ 80%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 เว็บเซอร์วิส

ผู้วิจัยได้ออกแบบระบบการจัดการบันทึกเวลาการเข้า – ออกงานด้วยสถาปัตยกรรมไมโครเซอร์วิสร่วมกับ RESTful APIs เพื่อให้ได้เว็บเซอร์วิสที่สามารถให้บริการการทำงานต่าง ๆ ได้ โดยจะนำ Swagger UI เพื่ออธิบาย APIs ในแต่ละเว็บเซอร์วิสของทุกไมโครเซอร์วิสดังนี้

### 1. APIs ส่วนของ User

users APIs for managing user from fingerprint devices	
GET	/users/
POST	/users/
GET	/users/id/{id}/
PUT	/users/id/{id}/
DELETE	/users/id/{id}/
PUT	/users/id/{id}/disable/
GET	/users/id/{id}/fingerprint/
PUT	/users/id/{id}/fingerprint/
DELETE	/users/id/{id}/fingerprint/
DELETE	/users/id/{id}/locations/name/{name}/
POST	/users/id/{id}/locations/name/{name}/
DELETE	/users/reset/

ภาพที่ 4.11 จอภาพ APIs ของไมโครเซอร์วิส User

### 2. ส่วนของ Internal-User

internal-users APIs for managing user on fingerprint machines	
GET	/internal-users/
GET	/internal-users/fingerprint/
PUT	/internal-users/id/{id}/
DELETE	/internal-users/id/{id}/
GET	/internal-users/id/{id}/
PUT	/internal-users/id/{id}/fingerprint/
DELETE	/internal-users/id/{id}/fingerprint/
GET	/internal-users/id/{id}/fingerprint/
POST	/internal-users/id/{id}/fingerprint/enroll/
DELETE	/internal-users/reset/

ภาพที่ 4.12 จอภาพ APIs ของไมโครเซอร์วิส Internal-User

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. ส่วนของ Attendance

attendances APIs about managing attendance of user	
GET	/attendances/
POST	/attendances/
GET	/attendances/id/{id}/
DELETE	/attendances/id/{id}/
DELETE	/attendances/reset/
GET	/attendances/status/{mode}/
GET	/attendances/tags/{name}/
GET	/attendances/time/
GET	/attendances/users/id/{id}/
DELETE	/attendances/users/id/{id}/
GET	/attendances/verified/{mode}/

ภาพที่ 4.13 จอภาพ APIs ของไมโครเซอร์วิส Attendance

### 4. ส่วนของ Location

locations APIs about getting and managing user & device location	
POST	/locations/
GET	/locations/
GET	/locations/name/{name}/
PUT	/locations/name/{name}/
DELETE	/locations/name/{name}/
POST	/locations/name/{name}/devices/serial/{sn}/
DELETE	/locations/name/{name}/devices/serial/{sn}/
POST	/locations/name/{name}/users/id/{id}/
DELETE	/locations/name/{name}/users/id/{id}/
DELETE	/locations/reset/

ภาพที่ 4.14 จอภาพ APIs ของไมโครเซอร์วิส Location

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5. ส่วนของ Device

**devices** APIs about get and managing device information

GET	/devices/
POST	/devices/
GET	/devices/active/
GET	/devices/inactive/
GET	/devices/ip/
PUT	/devices/ip/clearing-admin/
DELETE	/devices/locations/name/{name}/
DELETE	/devices/reset/
GET	/devices/serial/
DELETE	/devices/serial/{sn}/
GET	/devices/serial/{sn}/
PUT	/devices/serial/{sn}/
DELETE	/devices/serial/{sn}/locations/name/{name}/
POST	/devices/serial/{sn}/locations/name/{name}/

ภาพที่ 4.15 จอภาพ APIs ของไมโครเซอร์วิส Device

โดยความสมบูรณ์ของเว็บเซอร์วิสนั้นสามารถใช้ฟังก์ชันหลักได้แล้ว แต่ยังไม่ครบสมบูรณ์ตามความต้องการซึ่งจะถูกพัฒนาต่อไป โดยการพัฒนาเสร็จสิ้นไปแล้วอยู่ที่ประมาณ 90% เมื่อเทียบส่วนที่ใช้งานได้กับความต้องการ

## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

#### 5.1 สรุปผลการวิจัย

ผู้วิจัยได้ออกแบบระบบการจัดการบันทึกการลงเวลาการเข้า – ออกงานตามสถาปัตยกรรมไมโครเซอร์วิสโดยมีการติดต่อสื่อสารระหว่างกันโดยใช้ RESTful API ผู้วิจัยได้พัฒนาเว็บแอปพลิเคชันเป็นส่วนการติดต่อกับผู้ใช้งาน แต่ในส่วนของเว็บแอปพลิเคชันยังมีส่วนที่ยังพัฒนาไม่เสร็จสิ้น ต้องมีการพัฒนาต่อเพื่อให้ซอฟต์แวร์นี้สามารถนำไปทดแทนซอฟต์แวร์ตัวเก่าได้ ในส่วนที่พัฒนาเสร็จสิ้นและทำการทดสอบแล้วมีความเร็วในการตอบสนองที่ไวขึ้น

#### 5.2 ข้อเสนอแนะ

1. ระบบสามารถเพิ่มความสามารถในการเพิ่มข้อมูลสถานที่โดยใช้เทคโนโลยีของ Google Map เข้ามาเกี่ยวข้องได้ (ถ้าจำเป็น)
2. มีฟังก์ชันต้องแก้ไขเพื่อให้ได้ประสิทธิภาพที่ดีขึ้น เช่น การตรวจสอบว่าการลงทะเบียนลายนิ้วมือเสร็จสิ้นหรือไม่ เนื่องจากฮาร์ดแวร์ไม่มีการตอบสนองว่าการลงทะเบียนประสบความสำเร็จหรือไม่ จึงต้องมีการเขียนโปรแกรมลูปรตรวจสอบว่ามีลายนิ้วมือเพิ่มเข้ามาในอุปกรณ์หรือไม่ จึงทำให้อุปกรณ์ทำงานหนัก
3. การวิจัยนี้ควรมีผู้ร่วมวิจัยเนื่องจากเป็นระบบที่ต้องออกแบบและพัฒนาในหลายส่วน จะทำให้การวิจัยเสร็จสมบูรณ์เร็วยิ่งขึ้น

## บรรณานุกรม

- [1] What are microservices and why are they important. Retrieved July 26, 2019, from <https://www.brunton-spall.co.uk/post/2014/05/21/what-is-a-microservice-and-why-does-it-matter/>
- [2] Microservices. Retrieved July 26, 2019, from <https://en.wikipedia.org/wiki/Microservices>
- [3] ทำความรู้จักกับ Microservices. Retrieved July 26, 2019, from <https://www.techtalkthai.com/introduction-to-microservices-architecture/>
- [4] What are microservices. Retrieved July 27, 2019, from <https://microservices.io/>
- [5] Microservices vs. Monolith Architecture. Retrieved December 3, 2019, from [https://dev.to/alex\\_barashkov/microservices-vs-monolith-architecture-4l1m?source=post\\_page-----c84d996d062-----](https://dev.to/alex_barashkov/microservices-vs-monolith-architecture-4l1m?source=post_page-----c84d996d062-----)
- [6] Representational state transfer. Retrieved August 8, 2019, from [https://en.wikipedia.org/wiki/Representational\\_state\\_transfer](https://en.wikipedia.org/wiki/Representational_state_transfer)
- [7] REST กับ RESTful API ต่างกันนะรู้ยัง. Retrieved August 14, 2019, from <https://medium.com/@iamgique/restful-api-%E0%B8%81%E0%B8%B1%E0%B8%9A-rest-api-%E0%B8%95%E0%B9%88%E0%B8%B2%E0%B8%87%E0%B8%81%E0%B8%B1%E0%B8%99%E0%B8%99%E0%B8%B0%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%A2%E0%B8%B1%E0%B8%87-2c70c42990e3>
- [8] RESTful API Designing guidelines. Retrieved August 8, 2019, from <https://hackernoon.com/restful-api-designing-guidelines-the-best-practices-60e1d954e7c9>
- [9] What's the difference between REST & RESTful. Retrieved August 8, 2019, from <https://stackoverflow.com/questions/1568834/whats-the-difference-between-rest-restful>
- [10] zk-protocol. Retrieved July 8, 2019, from <https://github.com/adrobinoga/zk-protocol>