

ระบบบริหารจัดการประชุมทางวิชาการ
ACADEMIC CONFERENCE MANAGEMENT SYSTEM



ธนเกียรติ การุณ
ธาวิน บุญเชิญ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบบริหารจัดการประชุมทางวิชาการ

นายธนเกียรติ การุณ 59010555
นายชาวิน บุญเชิญ 59010651
อาจารย์วิบูลย์ พร้อมพานิชย์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2562

บทคัดย่อ

การประชุมทางด้านวิชาการ คือ การจัดงานที่ให้ผู้เขียนบทความทางด้านวิชาการได้มานำเสนอความรู้ที่ได้วิจัยและค้นพบ ผ่านการส่งบทความทางด้านวิชาการเข้ามาคัดเลือกโดยผู้จัดงาน ซึ่งเงื่อนไขของบทความที่ส่งมา จะต้องสอดคล้องกับหัวข้อที่ทางผู้จัดงานตั้งเอาไว้โดย มีกระบวนการทำงานตามลำดับคือการรับบทความ, การประเมินบทความ, การรับเลือกบทความ, การแจ้งผลการคัดเลือกบทความ, การลงทะเบียนและการชำระเงินค่าลงทะเบียน

โดยปฏิญานิพนธ์ฉบับนี้เป็นระบบบริหารจัดการประชุมทางวิชาการสำหรับสนับสนุนด้านการชำระเงินและลงทะเบียน โดยการชำระเงินสามารถทำได้หลายช่องทาง และมีรูปแบบส่วนลดต่าง ๆ แล้วแต่ผู้จัดงานต้องการ และระบบการออกใบเสร็จค่าลงทะเบียนสำหรับผู้เข้าร่วมงานอีกด้วย ผู้เข้าร่วมงานสามารถเข้าถึงระบบลงทะเบียนเพื่อสมัครหรือแก้ไขข้อมูลส่วนตัวได้ โดยผู้เข้าร่วมงานสามารถตรวจสอบสถานะการชำระเงินผ่านทาง Web Application ได้

ACADEMIC CONFERENCE MANAGEMENT SYSTEM

Thanakiat Karoon 59010555

Thawin Boonchoen 59010651

Wiboon Promphanich Advisor

Academic Year 2019

Abstract

Academic conference is an event that allow authors of academic article to have an opportunity for presenting and exchanging their research and discoveries through their academic article which have been selected by the conference's organizer. By attending to conference, every article must have been proceed a managing-conference processes such as paper submission system, paper evaluation system, register system and payment management system.

This thesis has develop to supports attendee and conference's organizer in registration's problem and financial management. Payment can be made via many methods with different discount form Depending on the conference's organizer with registration fee receipt system for attendees as well. Attendee can access the registration system to apply or edit personal information. Which the attendees can check the payment status via the Web Application

กิตติกรรมประกาศ

โครงการเรื่องระบบบริหารจัดการประชุมทางวิชาการจะไม่สามารถสำเร็จได้ ถ้าหากกลุ่มของข้าพเจ้าไม่ได้รับความช่วยเหลือ และ คำปรึกษาจากบุคคลเหล่านี้

บุคคลแรกที่ต้องกล่าวขอบคุณเป็นอย่างสูงคืออาจารย์ที่ปรึกษาโครงการของข้าพเจ้า อาจารย์วิบูลย์ พร้อมพานิชย์ ที่คอยให้คำปรึกษา แนะนำแนวทางของโครงการ ทิศทางการดำเนินงานต่างๆ

บุคคลถัดไปคือบุพการีของข้าพเจ้า ที่คอยเลี้ยงดู อบรมสั่งสอนและให้การสนับสนุนต่างๆ แก่ข้าพเจ้า พร้อมทั้งยังเป็นกำลังใจและให้คำปรึกษาในบางปัญหาที่ข้าพเจ้าไม่สามารถแก้ไขได้

และที่จะขาดไม่ได้เลยคือ คณะอาจารย์และบุคลากรต่างๆ ในสาขาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำ และสั่งสอนความรู้ต่างๆ มากมาย มาโดยตลอด ขอขอบคุณรุ่นพี่ และเพื่อนๆทุกคนในสาขาวิศวกรรมคอมพิวเตอร์ที่ให้คำแนะนำ ให้คำปรึกษา และเป็นกำลังใจเสมอมา จึงทำให้โครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

ธนเกียรติ การุณ

ธาวิน บุญเชิญ

สารบัญ

	หน้า
กิตติกรรมประกาศ.....	IV
สารบัญ.....	V
สารบัญรูปภาพ.....	VII
สารบัญตาราง.....	XI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.4 ข้อยกเว้นโครงการ.....	2
1.5 ขอบเขตโครงการ.....	2
1.6 แผนการดำเนินงาน.....	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1 Java Script Framework: React & Redux.....	4
2.2 ECMA Script: ES6.....	8
2.2.1 Syntax แบบใหม่.....	8
2.2.2 เพิ่ม Classes และ Modules.....	9
2.2.3 เพิ่มพีเจอรี่ใหม่.....	9
2.3 Microsoft SQL Server.....	10
2.3.1 System Database ของ Microsoft SQL Server.....	11
2.4 Payment Gateway.....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3 การออกแบบโครงงาน	13
3.1 ภาพรวมของระบบ	13
3.2 บัญชีผู้ใช้ภายในระบบ	14
3.3 Use Cases Diagram	15
3.4 การออกแบบฐานข้อมูล	20
3.4.1 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลงานประชุม	21
3.4.2 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลผู้ใช้ระบบ	23
3.4.3 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลผู้เข้าร่วมงานประชุม	25
3.5 Software Requirement Specification	26
บทที่ 4 ผลการดำเนินงาน	28
4.1 หน้าแรก	28
4.2 ระบบลงทะเบียนข้อมูลส่วนตัว	29
4.3 ระบบ Login เข้าสู่หน้าหลักของผู้ใช้งาน	30
4.4 ระบบสำหรับ Attendee	31
4.4.1 ระบบลงทะเบียนงานวิชาการ	31
4.4.2 ระบบสรุปผลการลงทะเบียน	32
4.4.3 ระบบชำระเงินค่าลงทะเบียน	33
4.4.4 ระบบแก้ไขข้อมูลสำหรับผู้เข้าร่วมงาน	33
4.5 ระบบสำหรับ Organizer	34
4.5.1 ระบบเพื่อดูภาพรวม	34
4.5.2 ระบบสร้างงานประชุม	35
4.5.3 ระบบแก้ไขงานประชุม	36

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการดำเนินงาน	28
4.5.4 ระบบตรวจสอบผู้เข้าร่วมงานประชุม	36
4.5.5 ระบบยืนยันการชำระเงินของผู้เข้าร่วมงาน	37
บทที่ 5 สรุปผลและข้อเสนอแนะ	38
5.1 สรุปผลการดำเนินงาน	38
5.2 แนวทางในการพัฒนาต่อ	39
ภาคผนวก ก. การติดตั้งเครื่องมือที่ใช้ในการพัฒนาระบบ	40
ภาคผนวก ข. Source Code	49
ภาคผนวก ค. คู่มือการใช้งานระบบ	66
บรรณานุกรม	82

สารบัญรูปภาพ

รูป	หน้า
รูป 2.1 เครื่องหมายการค้าของ React และ Redux	4
รูป 2.2 การเปรียบเทียบระหว่าง HTML กับ React(JSX).....	5
รูป 2.3 flow การทำงาน ของ Redux	6
รูป 2.4 ภาพแสดงเครื่องหมายของ ECMA Script 6 (ES6)	8
รูป 2.5 ภาพแสดงเครื่องหมายของ SQL Server.....	10
รูป 2.6 ตัวอย่างการใช้งาน Payment Gateway	12
รูป 3.1 ภาพรวมของระบบ Client-Server	13
รูป 3.2 ลำดับการเข้าถึงระบบของแต่ละบัญชี.....	14
รูป 3.3 แผนภาพ UML ของระบบ	15
รูป 3.4 การทำงานของ Database Config	16
รูป 3.5 การทำงานของ Database Editor	17
รูป 3.6 การทำงานของ Payment และ Receipt Release.....	18
รูป 3.7 การทำงานของ Create Conference	19
รูป 3.8 แผนภาพ ER Diagram ของการออกแบบฐานข้อมูล.....	20
รูป 4.1 หน้าแรกของระบบบางส่วน	28
รูป 4.2 ระบบลงทะเบียนข้อมูลส่วนตัวบางส่วน	29
รูป 4.3 ระบบ Login เข้าสู่หน้าหลักของผู้ใช้งาน	30
รูป 4.4 ระบบลงทะเบียนงานวิชาการบางส่วน	31
รูป 4.5 ระบบสรุปผลการลงทะเบียนบางส่วน	32
รูป 4.6 ระบบชำระเงินค่าลงทะเบียน	33
รูป 4.7 ระบบแก้ไขข้อมูลสำหรับผู้เข้าร่วมงาน ด้านข้อมูลส่วนตัว.....	33

สารบัญรูปภาพ (ต่อ)

รูป	หน้า
รูป 4.8 ระบบเพื่อคุณภาพรวม.....	34
รูป 4.9 ระบบสร้างงานประชุม	35
รูป 4.10 ระบบแก้ไขงานประชุม หน้าเล็องานประชุมที่ต้องการแก้ไข	36
รูป 4.11 ระบบตรวจสอบผู้เข้าร่วมงานประชุมบางส่วน	36
รูป 4.12 ระบบยืนยันการชำระเงินของผู้เข้าร่วมงาน	37
รูป 1ก. สัญลักษณ์ของ Semantic Ui React	41
รูป 2ก. สัญลักษณ์ของ Styled-Component	41
รูป 3ก. สัญลักษณ์ของ Material-Ui	42
รูป 4ก. สัญลักษณ์ของ npm	42
รูป 5ก. สัญลักษณ์ของ yarn	43
รูป 6ก. สัญลักษณ์ของ Swagger	43
รูป 7ก. สัญลักษณ์ของ My SQL	44
รูป 8ก. สัญลักษณ์ของ GitKraken	44
รูป 9ก. สัญลักษณ์ของ Visual Studio Code	45
รูป 10ก. สัญลักษณ์ของ ESLint	45
รูป 1ข. โครงสร้างของของมูลของโฟลเดอร์ Frontend	50
รูป 2ข. ไฟล์บางส่วนภายในโฟลเดอร์ assets	51
รูป 3ข. ตัวอย่างข้อมูลที่เก็บอยู่ใน Yarn.lock	51
รูป 4ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src	52
รูป 5ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components	52
รูป 6ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components/Attendee	53

สารบัญรูปภาพ (ต่อ)

รูป	หน้า
รูป 7ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components/Organizer	53
รูป 8ข. ตัวอย่างโค้ด app.js (Frontemd)	54
รูป 9ข. รูปตัวอย่างข้อมูลของ 1 Components	55
รูป 10ข. รูปตัวอย่างโค้ดของ index.js (Login).....	55
รูป 11 ข. รูปตัวอย่างโค้ดของ styled.js (Login).....	60
รูป 12 ข. โครงสร้างข้อมูลของโฟลเดอร์ Backend.....	61
รูป 13 ข. โครงสร้างข้อมูลของโฟลเดอร์ Backend/api.....	61
รูป 14 ข. ตัวอย่างโค้ด Backend (api/createTable)	62
รูป 15 ข. ตัวอย่าง โค้ด Backend ส่วนที่ 1	63
รูป 16 ข. ตัวอย่าง โค้ด Backend ส่วนที่ 2	63
รูป 17 ข. ตัวอย่าง โค้ด Backend ส่วนที่ 3	63
รูป 18 ข. ตัวอย่าง โค้ด Backend ส่วนที่ 4	63
รูป 19 ข. ตัวอย่าง โค้ด app.js (Backend).....	64
รูป 20 ข. ตัวอย่าง โค้ด server.js	65
รูป 1ค. หน้าจอการ log-in เข้าสู่ระบบ	67
รูป 2ค. หน้าจอการกรองข้อมูลส่วนตัว.....	68
รูป 3ค. หน้าจอการกรองข้อมูลสำหรับการออกไปเสร็จ	68
รูป 4ค. หน้าจอการเปลี่ยนรหัสผ่าน.....	69
รูป 5ค. หน้าจอลงทะเบียนงานประชุม (Active Conference).....	70
รูป 6ค. หน้าจอเปลี่ยนข้อมูล.....	71
รูป 7ค. หน้าจอเปลี่ยนข้อมูลส่วนตัว.....	71

สารบัญรูปภาพ (ต่อ)

รูป	หน้า
รูป 8ค. รูปแบบใบเสร็จงานวิชาการ	72
รูป 9ค. หน้าจอระบบชำระเงิน(Payment).....	73
รูป 10ค. หน้าจอระบบชำระเงินด้วยการโอนเงินผ่านธนาคาร	73
รูป 11ค. หน้าจอประวัติการลงทะเบียน (History)	74
รูป 12ค. หน้าจอภาพรวมของงานประชุม (Overview).....	75
รูป 13ค. หน้าจอระบบสร้างงานประชุม(Create Conference).....	76
รูป 14ค. หน้าจอระบบแก้ไขงานประชุม	77
รูป 15ค. หน้าจอระบบตรวจสอบข้อมูลผู้สมัคร(Attendee Data).....	78

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 1.1 แสดงแผนการดำเนินงาน	3
ตารางที่ 3.1 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลงานประชุม.....	21
ตารางที่ 3.2 คำอธิบายสำหรับส่วนเก็บข้อมูลงานประชุม.....	21
ตารางที่ 3.3 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ.....	23
ตารางที่ 3.4 คำอธิบายสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ.....	23
ตารางที่ 3.5 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลผู้เข้าร่วมงานประชุม	25
ตารางที่ 3.6 คำอธิบายสำหรับส่วนเก็บข้อมูลผู้เข้าร่วมงานประชุม	25
ตารางที่ 3.7 แสดงรายการความสามารถของระบบ	26

บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

การประชุมทางด้านวิชาการ คือ การจัดงานที่ให้ผู้เขียนบทความทางด้านวิชาการได้นำเสนอความรู้ที่ได้วิจัยและค้นพบ ผ่านการส่งบทความทางด้านวิชาการเข้ามาคัดเลือกโดยผู้จัดงาน ซึ่งเงื่อนไขของบทความที่ส่งมา จะต้องสอดคล้องกับหัวข้อที่ทางผู้จัดงานตั้งเอาไว้

โดยปกติแล้วจะมีการจัดงานประชุมทางด้านวิชาการระดับชาติขึ้นอยู่หลายงานต่อปี ไม่ว่าจะจัดภายในประเทศหรือต่างประเทศ ซึ่งผู้เข้าร่วมงานก็มีทั้งในและต่างประเทศเช่นเดียวกัน ซึ่งในปัจจุบันได้มีระบบต่าง ๆ ที่ช่วยให้ผู้เข้าร่วมงานสามารถลงทะเบียนเข้าร่วมงานได้ง่ายยิ่งขึ้น แต่ทว่าระบบเหล่านั้นยังไม่ครอบคลุมทุกกระบวนการโดยยังขาดระบบสำหรับลงทะเบียนเข้าร่วมงานและชำระเงินค่าลงทะเบียนออนไลน์

ทางผู้จัดทำได้สังเกตเห็นถึงปัญหาที่ผู้เข้าร่วมงานได้เผชิญอยู่บ่อยครั้ง ไม่ว่าจะเป็น ปัญหาการชำระเงินลงทะเบียนที่ยุ่งยาก, การชำระเงินในกรณีที่ต้องการส่งผลงานมากกว่า 1 งาน, ปัญหาในการออกใบเสร็จรับเงินในกรณีที่ต้องการเป็นชื่อของสถาบันอื่น เป็นต้น

จากปัญหาที่กล่าวมาข้างต้นทำให้ทางผู้จัดทำได้สังเกตเห็นว่า ถ้ามีระบบที่คอยเป็นตัวกลางในการจัดการประสานงานระหว่างผู้ใช้งานทั้ง 2 กลุ่ม คือ ผู้เข้าร่วมงานกับผู้จัดงานก็จะสามารถทำให้ช่วยลดปัญหาและขั้นตอนในการจัดการที่ยุ่งยากต่าง ๆ ได้ จึงได้ริเริ่ม “ระบบบริหารจัดการประชุมวิชาการ” นี้ขึ้นมา

โดยระบบนี้เป็นระบบที่ทำการต่อยอดมาจากระบบเก่าของรุ่นก่อนหน้าที่ยังสร้างระบบได้ไม่เสร็จสมบูรณ์

1.2 วัตถุประสงค์ของโครงการ

- 1) สร้างระบบขึ้นมาเพื่อสนับสนุนการจัดงานประชุมทางวิชาการ โดยทำหน้าที่เป็นตัวกลางในการจัดการประสานงานระหว่างผู้เข้าร่วมงานกับผู้จัดงานในด้านการลงทะเบียนและชำระเงิน
- 2) ช่วยเหลือผู้จัดงานในการแก้ปัญหาในการออกใบเสร็จให้มีประสิทธิภาพมากขึ้น
- 3) แก้ปัญหาการชำระค่าลงทะเบียนซ้ำของผู้เข้าร่วมงาน
- 4) เพื่ออำนวยความสะดวกในการชำระเงินได้หลากหลายช่องทางสำหรับผู้เข้าร่วมงานจากต่างประเทศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1) ช่วยอำนวยความสะดวกให้กับคือ ผู้เข้าร่วมงานกับผู้จัดงานในด้านชำระเงินและลงทะเบียนงาน
- 2) เป็นแนวทางให้สามารถนำไปเป็นระบบที่ใช้งานได้จริงและไปประยุกต์ใช้กับงานประชุมทางด้านวิชาการต่าง ๆ
- 3) ได้เรียนรู้จากการนำความรู้ที่เรียน มาประยุกต์ใช้เพื่อพัฒนาระบบได้จริง

1.4 ข้อจำกัดของโครงการ

- 1) ระบบของโครงการนี้เป็น Web Application ที่ครอบคลุมเฉพาะการลงทะเบียนงานและชำระเงินเท่านั้น
- 2) ผู้จัดงานจะต้องประเมินคุณสมบัติในการเข้าร่วมให้เสร็จสิ้นก่อน และทำการให้ข้อมูลผู้ที่ผ่านการคัดเลือกแก่ระบบเพื่อนำไปชำระเงินในภายหลัง
- 3) โครงการนี้ไม่ได้พัฒนาระบบชำระเงินขึ้นมาเอง แต่ใช้ระบบของทางธนาคาร, PayPal และโอนเงินผ่านบัญชีธนาคาร

1.5 ขอบเขตของโครงการ

- 1) สามารถลงทะเบียนเพื่อเข้าร่วมงานประชุมได้
- 2) สามารถแก้ไขข้อมูลการลงทะเบียนได้
- 3) สามารถตรวจสอบสถานะการชำระเงินค่าลงทะเบียนของผู้เข้าร่วมได้
- 4) สามารถชำระเงินค่าลงทะเบียนได้
- 5) สามารถพิมพ์ใบเสร็จหลังจากชำระเงินค่าลงทะเบียนเสร็จสิ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 แผนการดำเนินงาน

โดยแผนการดำเนินงานทั้งหมดทั้ง 2 เทอมจะถูกแบ่งอยู่ในตารางดังนี้

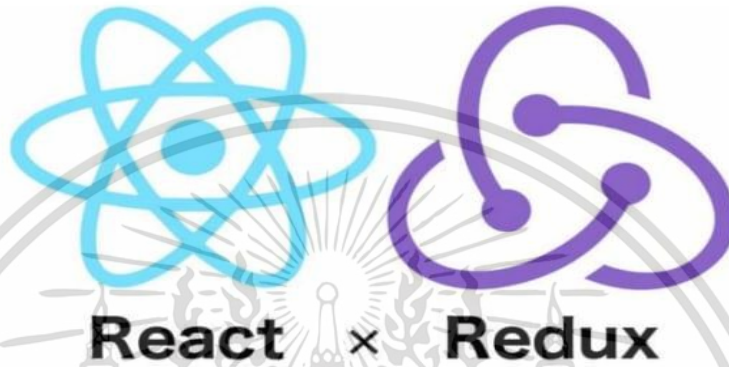
ตารางที่ 1.1 แสดงแผนการดำเนินงาน

เดือน	ส.ค	ก.ย	ต.ค	พ.ย	ธ.ค	ม.ค	ก.พ	มี.ค
แผนการดำเนินงาน								
ทำการศึกษาระบบก่อนหน้านี้ และทราบถึงปัญหาที่ต้องแก้ไข								
ทำการแก้ไขหน้า UI								
ทำการแก้ไขระบบลงทะเบียน								
ทำการแก้ไขระบบชำระเงินผ่านธนาคาร								
ทำการสร้างระบบสำหรับผู้จัดงานและระบบผู้เข้าร่วมงาน								
ทำการสร้างระบบสร้างการประชุมวิชาการ								
ทำการทดสอบระบบและแก้ไขส่วนที่ผิดพลาด								

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 Java Script Framework: React & Redux



รูป 2.1 เครื่องหมายการค้าของ React และ Redux

React คือ JavaScript Library ที่ทีม Facebook เป็นคนพัฒนาขึ้นมา และเปิดให้คนทั่วไปนำมาใช้ฟรี ซึ่งเว็บไซต์ในปัจจุบันของ Facebook.com ก็ใช้ React ในการพัฒนาด้วยเช่นกัน โดยการ จัดหน้า View ด้วย React จะมีอยู่ 3 Concept คือ

1) **Component** (ส่วนประกอบ)

React จะมองส่วนต่าง ๆ ในหน้าเว็บเป็น Component ทั้งหมด เพื่อรองรับการ Reuse โดยที่เราไม่ต้องมาเขียน code ใหม่ เช่น button, header, footer, ... เป็นต้น

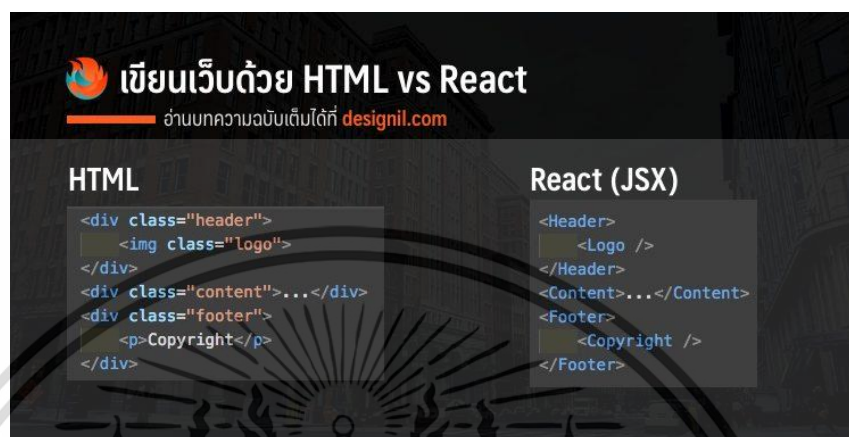
2) **State**

React จะเก็บข้อมูลต่าง ๆ ที่อยู่ใน Component นั้น ๆ ไว้ที่ State หรือกล่าวได้ว่า State คือ ตัวเก็บข้อมูลของ Component นั้น ๆ

3) **Props (Properties)**

เป็นการกำหนดคุณสมบัติให้กับ Component นั้น ๆ เพื่อใช้ส่งข้อมูลจาก Component หนึ่ง ไปให้อีก Component หนึ่ง

สำหรับการเขียน Component นั้นจะเหมือนกับการเขียน HTML โดย React ใช้สิ่งที่เรียกว่า **JSX** ในการแสดงผลเว็บไซต์ โดยจะมีรูปแบบการเขียนเหมือน HTML มาก แต่แตกต่างกันตรงที่เขียนเข้าไปในไฟล์ JavaScript แทนไฟล์ HTML



รูป 2.2 การเปรียบเทียบระหว่าง HTML กับ React(JSX)

สาเหตุที่ผู้จัดทำเลือก React ในการพัฒนาระบบเพราะเป็น Framework ที่ลดภาระการทำงานของ Server ได้เช่น ปกติในการจำกัดสิทธิ์เข้าถึงส่วนต่าง ๆ ของ Website หากเขียนด้วย HTML จำเป็นต้องทำการตรวจกับ Server ทุกครั้งที่ทำการ request แต่ถ้าใช้งาน React จะสามารถตรวจสอบได้ตั้งแต่ Front-End โดยไม่ต้องผ่าน Server โดยจะทำการตรวจสอบ URL ว่ามีสิทธิ์เข้าถึงได้หรือไม่ เป็นต้น และ React ยังสามารถ Render ได้ทั้งฝั่ง Server และ Client จึงทำให้ React สามารถเขียนเว็บแบบ “Isomorphic JavaScript” หรือ “Universal JavaScript” ได้ ซึ่งหมายถึงสามารถเขียนโค้ดด้วยภาษา JavaScript ทั้งฝั่ง Server และ Client

Redux คือ เป็นไลบรารีตัวหนึ่งที่ช่วยคุณให้การไหลเวียนข้อมูลภายใน React ให้ดีขึ้น เพื่อให้เราสามารถคาดการณ์ได้ว่าเมื่อเกิดเหตุการณ์หนึ่งใน Components ของเราแล้วจะมีผลกระทบต่อ Application ของเราอย่างไร โดยมี Principle 3 ข้อของ Redux ที่ต้องเข้าใจ ประกอบไปด้วย

1) Single source of truth

หมายความว่า State ต่าง ๆ ของระบบงานจะถูกจัดเก็บไว้เพียงที่เดียวเท่านั้น โดยที่จัดเก็บจะเรียกว่า **Store** อยู่ในรูปแบบของ object tree

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) State is read-only

สถานะต่าง ๆ ไม่สามารถเปลี่ยนแปลงได้ดังนั้นเมื่อเกิดเหตุการณ์ หรือ **Action** ขึ้นมาจะทำการสร้าง State ใหม่ขึ้นมาต่อจาก State เดิมโดยไม่ทำการแก้ไข State เดิม ทำให้เราสามารถจัดเก็บ State และ Replay ได้อีกด้วย นั่นคือ เราสามารถ recovery state ต่าง ๆ ได้รวมทั้งง่ายต่อการทดสอบอีกด้วย

3) Changes are made with pure functions

เมื่อมีเหตุการณ์ขึ้นมาแล้ว State ต่อไป นั้นเป็นหน้าที่ของ **Reducer** ซึ่ง Reducer นั้นจะเป็น **pure function** โดยพิจารณาจาก State ปัจจุบัน + Action ทำให้จะได้ State ต่อไปขึ้นมา ซึ่ง Reducer จะส่ง State ใหม่ออกมาเท่านั้น โดยไม่เปลี่ยนแปลง State ก่อนหน้าโดยเด็ดขาด โดยการทำงานต่าง ๆ ของ Redux จะประกอบไปด้วย

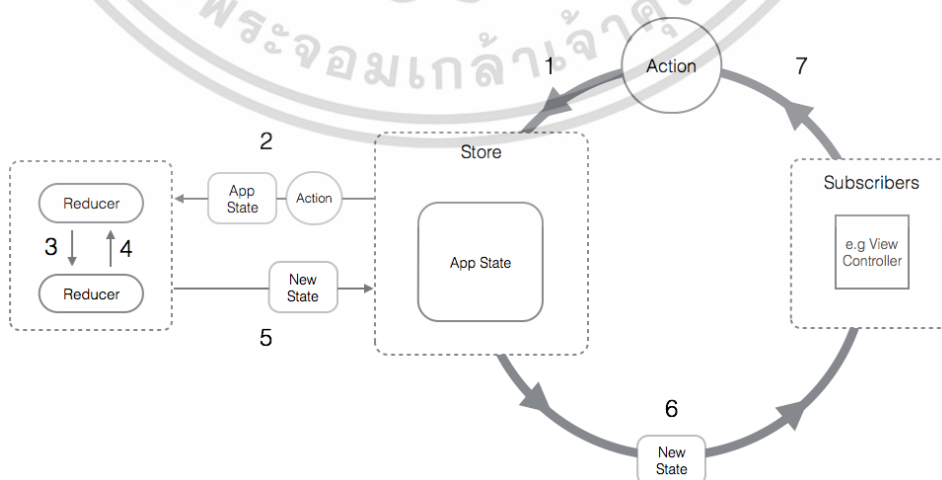
State คือ สถานะต่าง ๆ ของการทำงาน

Action คือ เหตุการณ์ต่าง ๆ ที่เกิดขึ้น

Store คือ ที่จัดเก็บ State ต่าง ๆ หรือสามารถเรียกว่าเป็น State Container และ notify ไปยังส่วนต่าง ๆ ที่ subscribe ไว้

Reducer คือ pure function ที่ทำการสร้าง State ถัดไปโดยพิจารณาจาก State ปัจจุบัน และ Action ที่เกิดขึ้น

ขั้นตอนการทำงานของ **Redux** จะเป็นแบบ **Unidirectional data flow** หรือการทำงานแบบทางเดียว ไม่มีย้อนกลับ แสดงดังรูป



รูป 2.3 flow การทำงาน ของ Redux

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เมื่อเกิด Action ต่าง ๆ ขึ้นมา ไม่ว่าจะเกิดจาก view/data source หรืออะไรก็ตามที่ต้องการเปลี่ยนแปลง State จะถูกส่งไปที่ Store
 2. จากนั้น Store จะทำการเรียกใช้งาน Reducer โดยส่ง State ปัจจุบัน และ Action ไปให้
 3. ทำการเรียก Reducer เพื่อทำงานตาม Action
 4. จากนั้น Reducer ทำการส่ง State ใหม่กลับมา
 5. ส่ง State ใหม่กลับไปให้ Store
 6. จากนั้น Store จะทำการ notify หรือแจ้งไปยังส่วนการทำงานต่าง ๆ ที่ subscribed ไว้เพื่อทำการ update ต่อไป เช่น การแสดงผล เป็นต้น
 7. ส่วนการแสดงผล หรือ data source อาจจะสร้าง Action ใหม่ ๆ ขึ้นมาได้อีก เช่น จากผู้ใช้งาน และ ข้อมูลจากภายในระบบ เป็นต้น
- และทำแบบนี้วนไปเรื่อย ๆ ในทุก ๆ Action



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ECMA Script: ES6



รูป 2.4 ภาพแสดงเครื่องหมายของ ECMA Script 6 (ES6)

ECMA Script คือมาตรฐานของภาษา JavaScript ที่กำหนดโดยองค์กร ECMA International เพื่อให้ภาษา Script นั้นมีความเป็น Standard นั่นก็คือการเปลี่ยน syntax และเพิ่ม feature ใหม่ ๆ ทั้งนี้ก็เพื่อที่จะรองรับการเขียน Application ที่มีความซับซ้อนมาก ๆ โดยมีคุณสมบัติและความสามารถดังนี้

2.2.1 Syntax แบบใหม่

- **Let + Const** โดย let จะเอาไว้สร้างตัวแปรเหมือนกับ var เลยครับ เพียงแต่ตัวแปรที่สร้างด้วย let นั้นจะมี scope เพียงแค่ภายใน block ส่วน const นั้นจะเหมือนกับ let แต่จะไม่สามารถเปลี่ยนแปลงค่าได้
- **Default Parameters** โดยสามารถกำหนด default parameter ต่างๆ ให้กับ function ได้ ซึ่งจะทำให้ไม่จำเป็นต้องเช็คค่า parameter นี้มีค่าหรือไม่

2.2.2 เพิ่ม Classes และ Modules

- **Classes** สามารถเขียน js แบบเป็น class ได้เหมือนกับ PHP หรือ Java มี constructor, สามารถ extends class อื่นๆ ได้ และมี static method
- **Modules** สามารถเขียนโค้ดแบบแบ่งเป็น module ได้ โดยสามารถเขียนโค้ดแยกเป็นไฟล์ของแต่ละ module ได้ และเมื่อต้องการใช้ module ไหน ค่อยไป import ไฟล์ module นั้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 เพิ่มฟีเจอร์ใหม่

- **Set** เป็นการเก็บข้อมูลแบบใหม่ที่เรียกว่า set โดยจุดเด่นคือถึงแม้ว่าใน set จะมี member ที่ซ้ำกันบางตัว แต่พอแสดงค่าของ set ออกมาก็จะมีเฉพาะ member ที่ไม่ซ้ำกัน
- **Map** คล้ายคลึงกับ set แต่จะมี member เป็น key/value
- **Promises** เอาไว้รัน โค้ดแบบ asynchronous โดยสามารถกำหนดได้ว่าจะให้ทำอะไร ถ้าทำสำเร็จให้ทำอะไร ถ้าล้มเหลวให้ทำอะไร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Microsoft SQL Server



รูป 2.5 ภาพแสดงเครื่องหมายของ SQL Server

Microsoft SQL Server เป็น โปรแกรมในการบริหารจัดการฐานข้อมูลที่ดีที่สุดในของ Microsoft โดยเป็นในรูปแบบของ Relational Database Management System หรือที่เรียกว่า RDBMS ซึ่งจะ เป็นบริหารข้อมูลให้กับผู้ใช้บริการต่าง ๆ รองรับการทำงานได้จำนวนมาก โดยรันอยู่บน Window NT ซึ่งใช้ภาษา T-SQL ในการดึงเรียกข้อมูล ด้วยเหตุที่ข้อมูลส่วนใหญ่ทั่วโลกเก็บไว้ในเครื่องที่ใช้ Microsoft Windows เป็น Operation System จึงทำให้เป็นการง่ายต่อ Microsoft SQL ที่จะนำข้อมูลที่ อยู่ในรูป Windows Based มาเก็บและประมวลผล และมีความสามารถมากมาย เทียบเท่ากับระบบ ฐานข้อมูลอื่น ๆ เช่น Oracle, DB2, Informix เป็นต้น

ภายใต้การจัดการของ Microsoft SQL Server นั้น จะมีฐานข้อมูลที่ Microsoft SQL Server จะต้องใช้ และมีมาตั้งแต่ต้น นับตั้งแต่ Install กันเลย โดยฐานข้อมูลนี้จะอยู่ในกลุ่มของ System Database ซึ่งแต่ละฐานข้อมูลจะมีหน้าที่ในการทำงานต่าง ๆ กันดังนี้

2.3.1 System Database ของ Microsoft SQL Server

Master Database

มีความสำคัญมากที่สุด ใช้สำหรับเก็บข้อมูลที่สำคัญของระบบ เช่น Meta Data พวก User, Login Information, Error Message, Linkedserver รวมถึงบอกตำแหน่งใน Primary File ในแต่ละข้อมูลอีกด้วย โดยหากฐานข้อมูลนี้มีปัญหา ก็จะส่งผลกระทบต่อการทำงานของ Microsoft SQL Server เลยทีเดียว

MSDB Database

สำคัญ รองจาก Master เก็บข้อมูลเกี่ยวกับ Alert, Job, Schedule ซึ่งถูกใช้โดยบริการของ SQL Server Agent ซึ่งเป็นบริการที่ทำงานอัตโนมัติ เช่น Backup ข้อมูลอัตโนมัติแต่ละวัน การแจ้งเตือนเมื่อระบบมีปัญหา โดยจะทำการส่ง Email ให้กับผู้ดูแลระบบ เป็นต้น

Distribution Database

ใช้ในการ Replication ของฐานข้อมูล เพื่อให้ข้อมูลของ Microsoft SQL Server แต่ละที่มีความตรงกัน

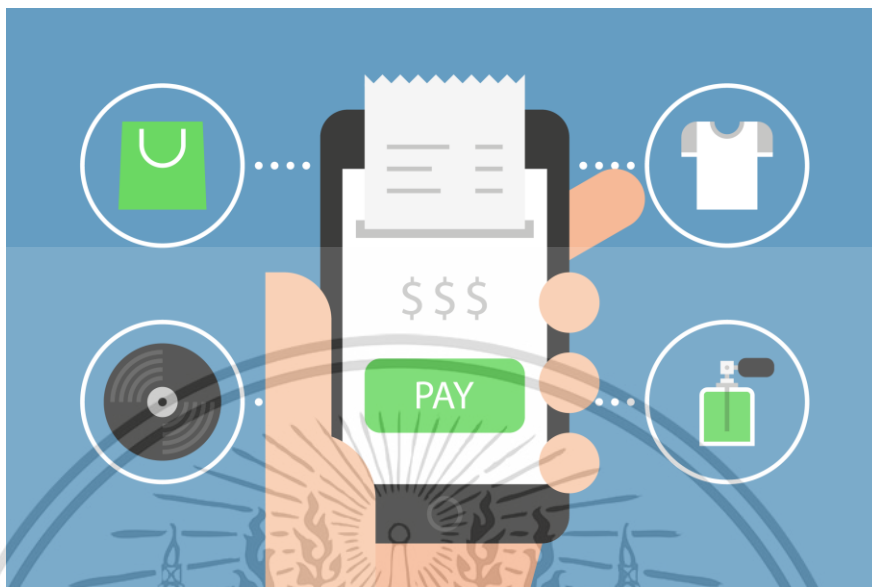
Model Database

เป็นฐานข้อมูลที่เป็นต้นแบบ (Database Template) กรณีที่เราสร้าง Database ใหม่ Microsoft SQL Server จะเอา Model Database นี้มาเป็นตัวตั้งต้น

TempDB Database

ไว้ในการเก็บข้อมูลที่เป็นชั่วคราว Temporary สำหรับ process ที่จำเป็นต้องการนำ data มาพักไว้ก่อนแล้วค่อยนำไปทำอย่างอื่นต่อ ลบทุกครั้งที่มีการ Restart Service หรือ Shutdown ระบบ

2.4 Payment Gateway



รูป 2.6 ตัวอย่างการใช้งาน Payment Gateway

Payment Gateway คือช่องทางการชำระเงินออนไลน์ ที่ทำให้เราสามารถชำระค่าบริการต่าง ๆ ได้อย่างรวดเร็ว และปลอดภัย ช่วยลดเวลาในการเลือกซื้อสินค้าบนเว็บไซต์ โดยปัจจุบันมีผู้ให้บริการอยู่ 2 รูปแบบคือ การชำระเงินออนไลน์โดยเชื่อมกับธนาคารโดยตรง (Bank) อย่างเช่น Merchant iPay ของธนาคารกรุงเทพ และการชำระเงินออนไลน์ผ่านตัวกลางที่ไม่ใช่ธนาคาร อย่าง Paypal และ GB Prime Pay

Payment Gateway Bank ระบบชำระเงินออนไลน์ที่เชื่อมต่อกับธนาคารโดยตรง จะช่วยให้ธุรกิจ และเว็บไซต์ของคุณมีความน่าเชื่อถือมากยิ่งขึ้น มีความปลอดภัย อีกทั้งยังมีค่าธรรมเนียมต่อรายการที่ค่อนข้างต่ำ แต่การเชื่อมต่อกับธนาคารโดยตรง ทำให้มีเงื่อนไขในการขอใช้งานเยอะ ต้องมีเงินฝากค้ำประกัน โดยค่าตัวจะอยู่ที่ประมาณ 100,000 บาท ทำให้ Payment Gateway รูปแบบนี้เหมาะกับบริษัท หรือห้างหุ้นส่วนมากกว่า เนื่องจากมีข้อจำกัดด้านทุนจดทะเบียนเข้ามาเกี่ยวข้อง ด้วย ส่วน Payment Gateway Non-Bank เหมาะกับธุรกิจขนาดเล็ก ใหญ่ ไปจนถึงห้างหุ้นส่วน สามารถติดตั้งได้ง่ายๆ แถมไม่ต้องมีเงินฝากค้ำประกัน แต่ค่าธรรมเนียมต่อรายการนั้นจะสูงกว่าประเภท Bank

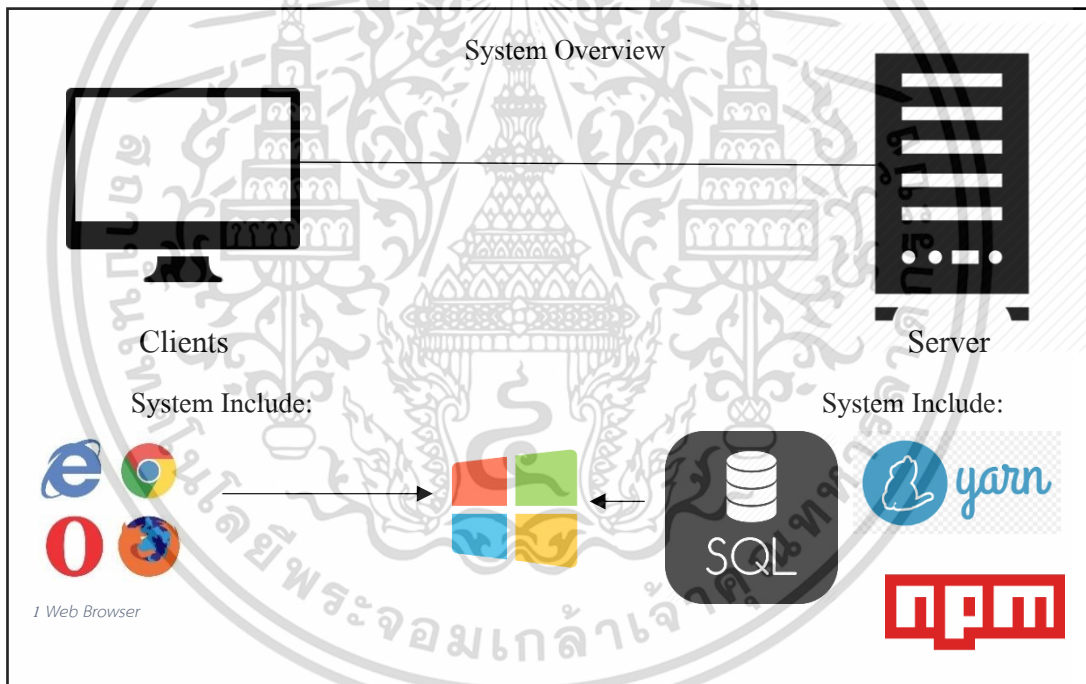
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบโครงงาน

การออกแบบจะเป็นการสร้างระบบกลางขึ้นมาเพื่อรองรับกระบวนการทั้งหมดในการลงทะเบียนของงานประชุมทางวิชาการ โดยระบบที่สร้างจะมีหน้าที่คือ การลงทะเบียนเพื่อชำระเงิน หลังจากที่ทางผู้จัดงานได้ส่งข้อมูลรายชื่อของผู้ที่ผ่านการคัดเลือกมาแล้วให้กับทางระบบเป็นที่เรียบร้อย จากนั้นจะนำข้อมูลเข้ามาไว้ยังฐานข้อมูลของระบบแล้วให้ผู้เข้าร่วมงานได้ชำระเงินผ่านระบบที่เราได้พัฒนาขึ้น โดยระบบจะสามารถจัดการและติดตามสถานะต่าง ๆ ให้กับทางผู้จัดการและผู้เข้าร่วมงาน รวมถึงปัญหาต่าง ๆ ด้วยเช่นกัน

3.1 ภาพรวมของระบบ



รูป 3.1 ภาพรวมของระบบ Client-Server

โดยระบบที่ออกแบบขึ้นมาทำงานอยู่ในระบบ Client-Server ในฝั่งของ Client นั้นจะเป็นส่วนของผู้ใช้งานทั้งหมด ซึ่งผู้ใช้งานจะสามารถใช้ระบบปฏิบัติการใดก็ได้ โดยใช้งานผ่าน Web Browser ซึ่งรองรับการใช้งานสำหรับ Web Browser แบบ Full-Screen ในคอมพิวเตอร์เท่านั้น

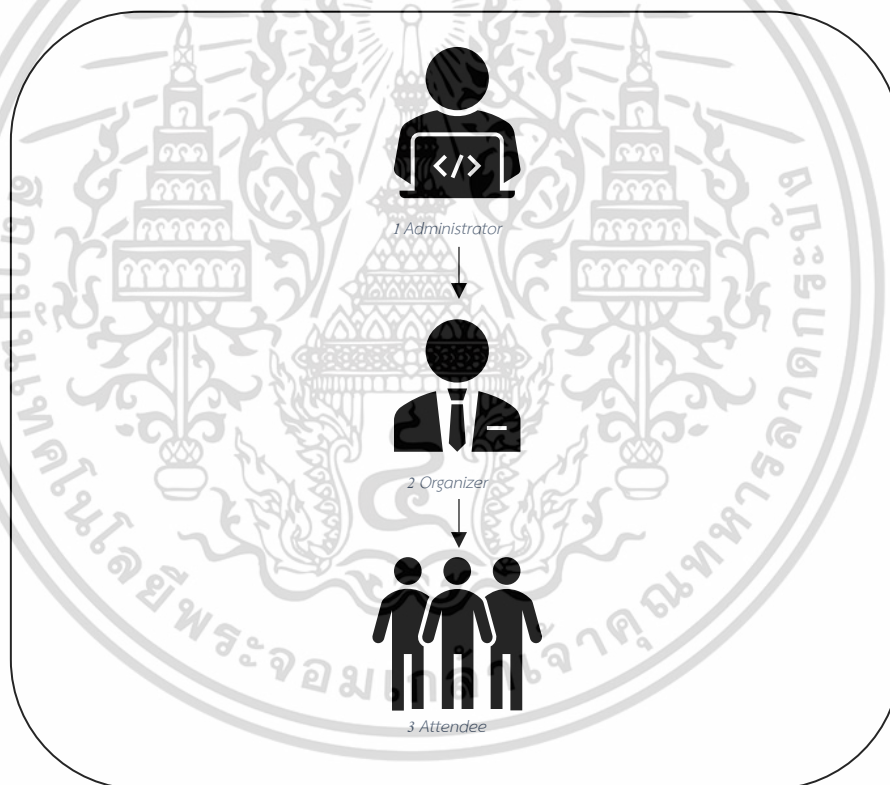
สำหรับฝั่ง Server นั้นจะใช้ระบบปฏิบัติการ Windows ซึ่งภายใน Server นั้น ได้ติดตั้งไฟล์โปรเจกต์ของระบบเอาไว้ และมีการติดตั้ง SQL Database ของระบบ โดยจำเป็นต้องติดตั้ง Node Package และ Node.js ไว้ด้วยเพื่อเชื่อมต่อส่วน Front-End เข้ากับส่วน Back-End ของระบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 บัญชีผู้ใช้ภายในระบบ

โดยระบบจะมีบัญชีผู้ใช้ทั้งหมด 3 แบบ โดยแต่ละบัญชีจะมีสิทธิ์ในการเข้าถึงระบบได้แตกต่างกัน ซึ่งจะแบ่งออกเป็นลำดับขั้นดังนี้

- **Administrator** หรือ System Admin ที่มีสิทธิ์ในการปรับแต่งระบบทั้งหมด โดยสามารถปรับเปลี่ยนและจัดการข้อมูลต่างๆทั้งแก้ไข ลบ และเพิ่ม
- **Organizer** หรือผู้จัดงาน โดยระบบจะกำหนดให้มีเพียง 1 บัญชีซึ่งเป็นของหน่วยงานที่จัดงานเท่านั้น โดยบัญชีนี้จะสามารถปรับเปลี่ยนข้อมูลทั้งหมดของงานประชุมวิชาการที่ตนเองจัดขึ้นได้
- **Attendee** หรือผู้เข้าร่วมงาน เป็นบัญชีสามารถตรวจสอบหรือแก้ไขเอกสารที่ตนเองได้ส่งเข้าระบบได้ รวมถึงการชำระเงินค่าเข้าร่วมงานประชุมวิชาการต่างๆที่ตนเองต้องการจะเข้าร่วม

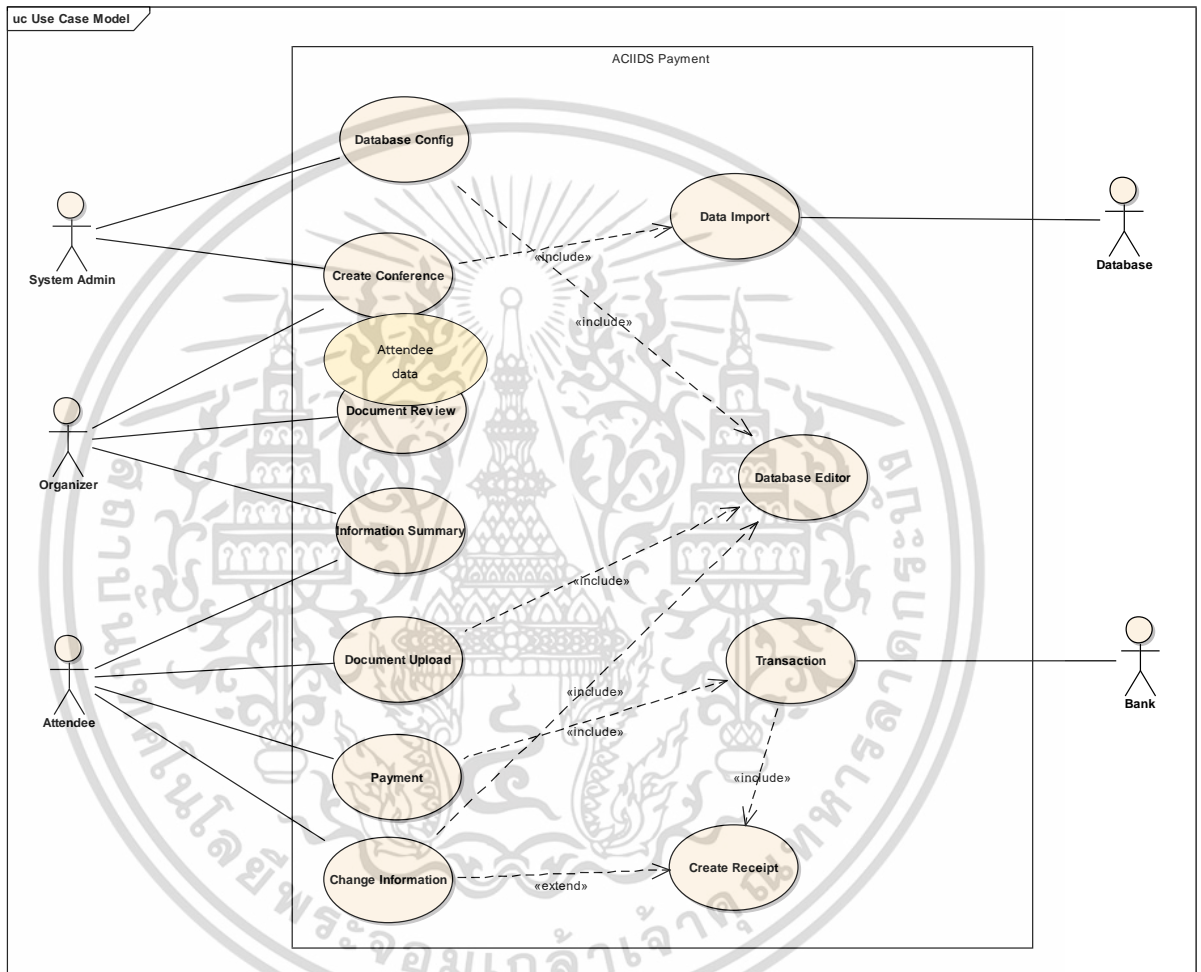


รูป 3.2 ลำดับการเข้าถึงระบบของแต่ละบัญชี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Use Cases Diagram

จากที่ผ่านมาเป็นการแสดงหัวข้อหลักสำหรับฟังก์ชันการทำงานของระบบ แต่ยังไม่ทราบอย่างแน่ชัดว่าในระบบมีไว้สำหรับผู้ใช้ประเภทใดบ้างและมีความสัมพันธ์ระหว่างฟังก์ชันอย่างไร ในส่วนนี้จะแสดงความสัมพันธ์ระหว่างผู้ใช้งาน, การติดต่อกับระบบภายนอก และฟังก์ชันการทำงาน ดังแผนภาพต่อไปนี้

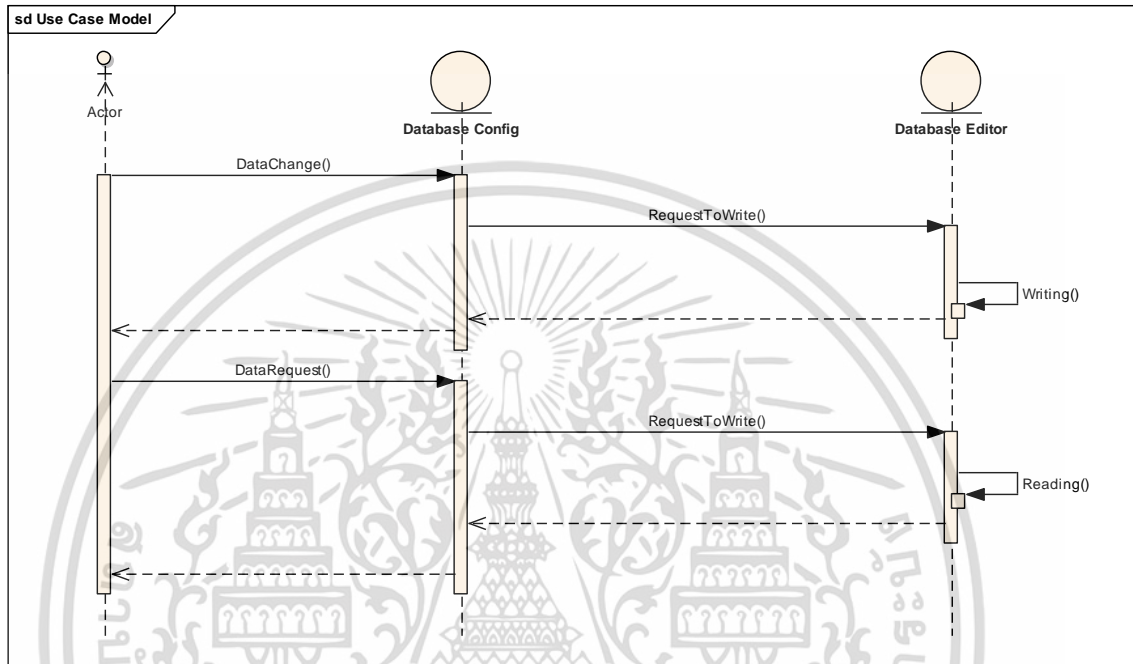


รูป 3.3 แผนภาพ UML ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะแบ่งฟังก์ชันการทำงานหลักๆเป็นดังนี้

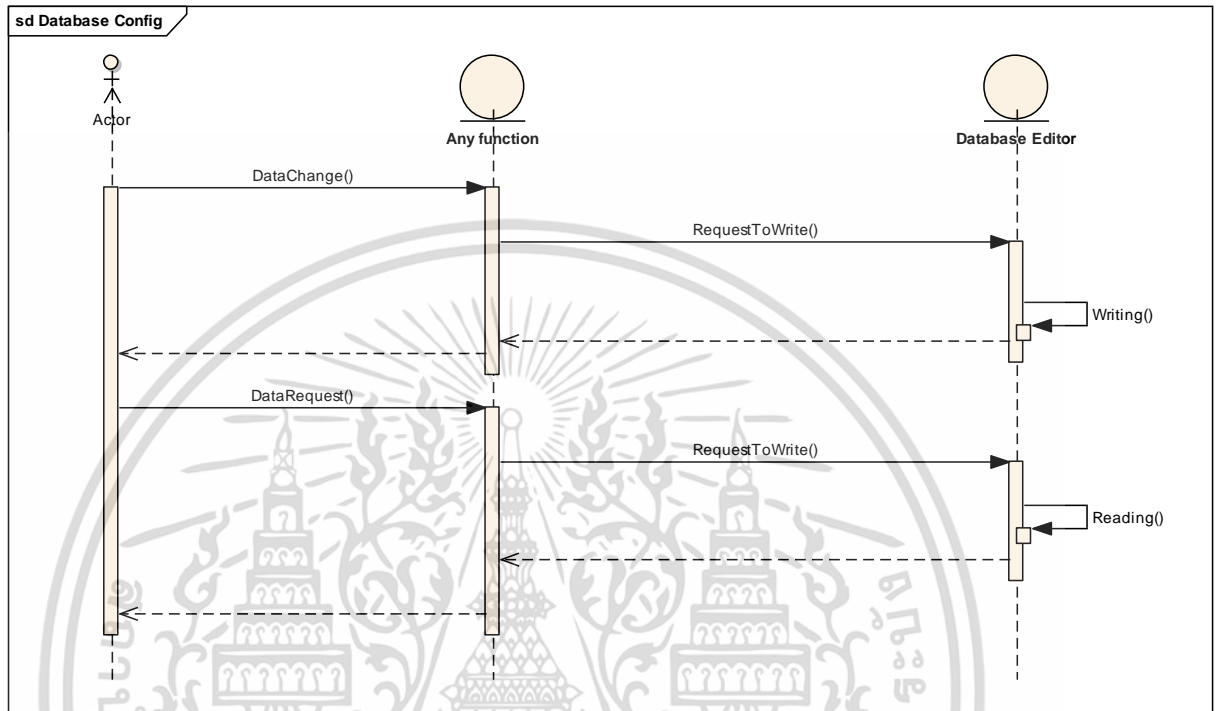
- ระบบจัดการฐานข้อมูล (Database Config) เป็นระบบที่ผู้ดูแลระบบเข้าไปจัดการกับฐานข้อมูลซึ่งสามารถอ่านและแก้ไขข้อมูลได้ โดยระบบจะทำการไปเรียกใช้งานในส่วนที่แก้ไขฐานข้อมูลอีกครั้งเมื่อมีการขอเขียนข้อมูลหรืออ่านข้อมูล ดังนี้



รูป 3.4 การทำงานของ Database Config

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

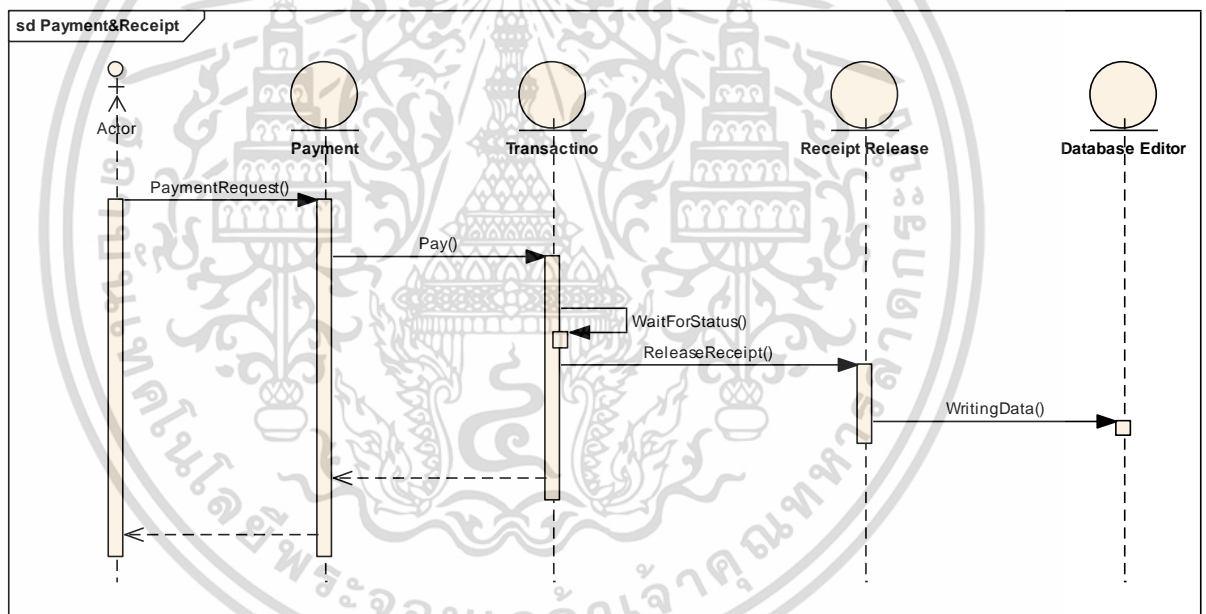
- ระบบแก้ไขฐานข้อมูล (Database Editor) เป็นระบบที่ใช้ในการเขียนและอ่านข้อมูลของระบบ โดยส่วนนี้จะถูกเรียกใช้ก็ต่อเมื่อมีการเปลี่ยนแปลงของข้อมูลหรือสถานะต่าง ๆ ในฐานข้อมูล โดยการทำงานของฟังก์ชันนี้จะเป็นตัวกลางในการแก้ไขข้อมูลต่าง ๆ ของทุกฟังก์ชัน ดังนี้



รูป 3.5 การทำงานของ Database Editor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

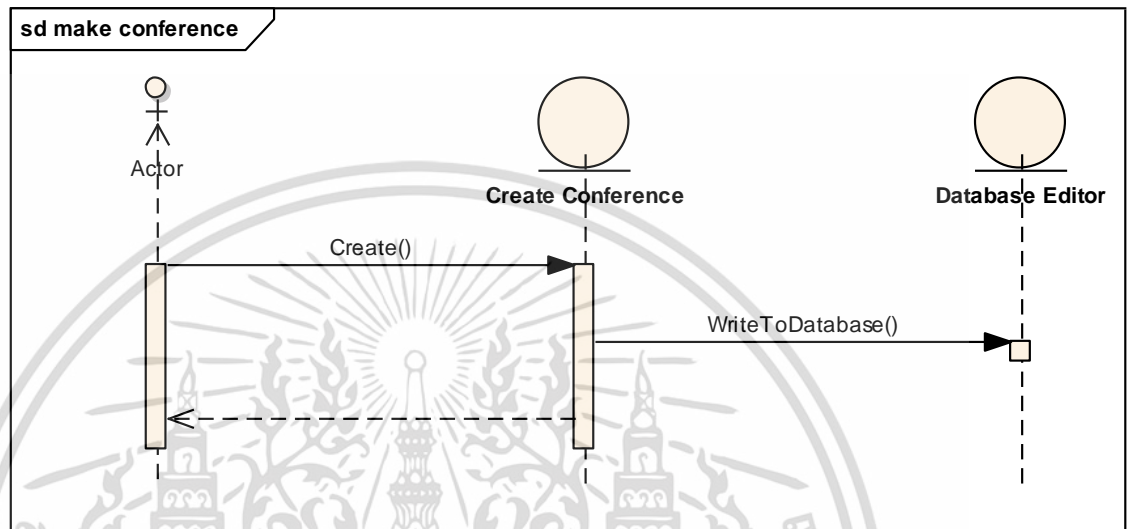
- ระบบแก้ไขข้อมูล (Change Information) ใช้สำหรับให้ผู้เข้าร่วมงานสามารถแก้ไขข้อมูลส่วนตัว ข้อมูลใบเสร็จ และอื่นๆได้ โดยการทำงานจะมีลักษณะที่คล้ายคลึงกับฟังก์ชัน Database Config
- ระบบชำระเงิน (Payment) เป็นฟังก์ชันสำหรับการชำระเงินค่าลงทะเบียน โดยจะมีการเรียกใช้งานระบบ Transaction หรือ ระบบรายการเดินบัญชี สำหรับเก็บประวัติการชำระเงินและรวบรวมข้อมูลให้กับระบบภายนอก และจะมีการเรียกใช้งานระบบใบเสร็จรับเงิน (Receipt Release) ในตอนท้ายด้วย
- ระบบออกใบเสร็จรับเงิน (Receipt Release) ทำหน้าที่ออกใบเสร็จหลังจากทำการชำระเงินแล้ว โดยผู้ใช้งานระบบสามารถปรับเปลี่ยนข้อมูลได้ตามเงื่อนไข



รูป 3.6 การทำงานของ Payment และ Receipt Release

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

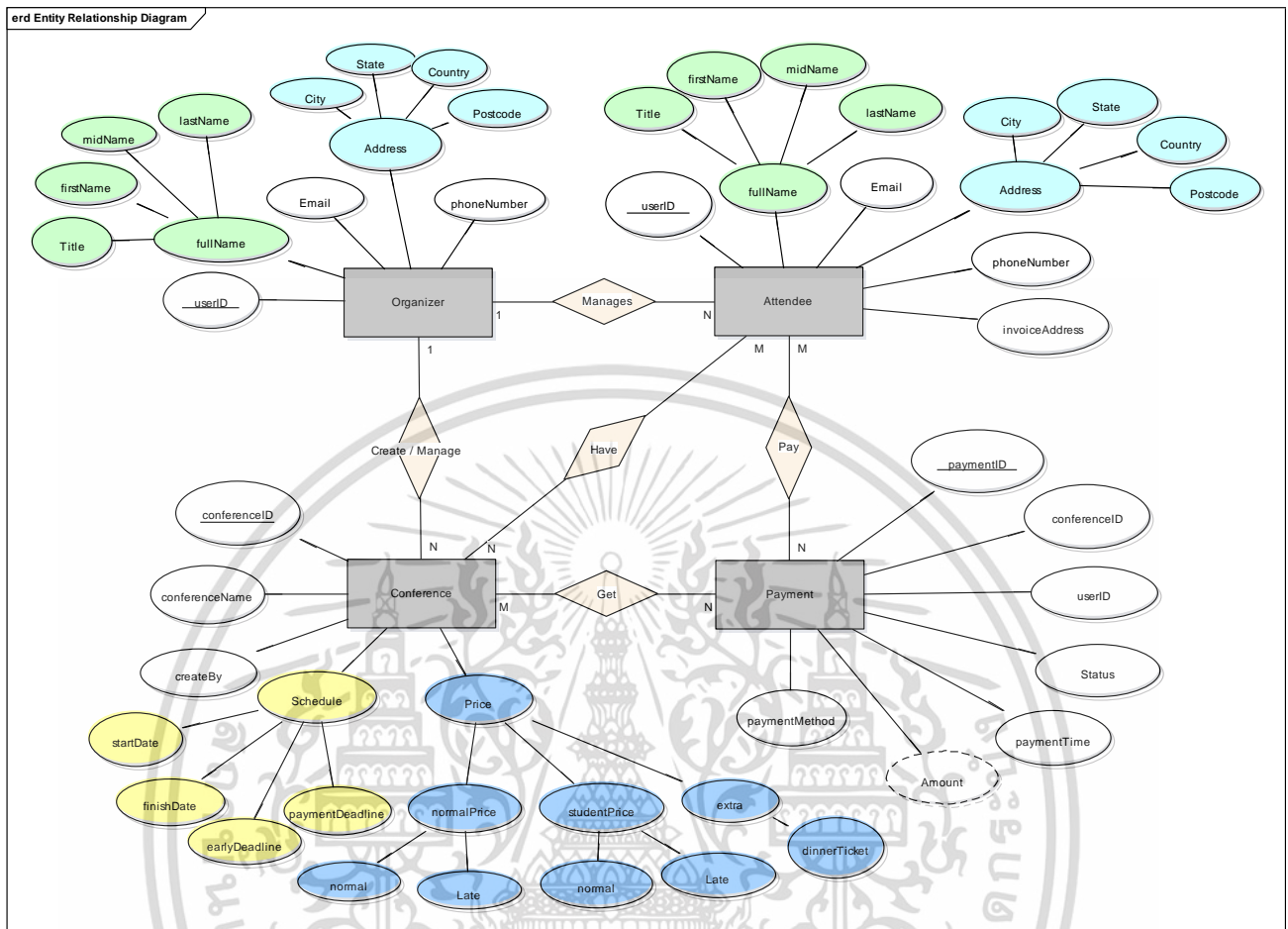
- ระบบการสร้างงานประชุม (Create Conference) โดยจะเป็นส่วนที่สามารถใช้ได้เฉพาะผู้จัดงาน ซึ่งใช้สร้างงานประชุม โดยสามารถกำหนดรายละเอียดต่าง ๆ ของงานประชุมได้ เช่น วันที่จัดงาน, อัตราค่าลงทะเบียน เป็นต้น โดยจะมีการไปเรียกใช้งานฟังก์ชัน Database Editor เพื่อเขียนข้อมูลลงระบบ



รูป 3.7 การทำงานของ Create Conference

- ระบบตรวจสอบข้อมูลผู้เข้าสมัคร (Attendee Data) ระบบนี้ จะให้ผู้จัดงานสามารถเข้าไปตรวจสอบข้อมูลของผู้สมัครรวมถึงแก้ไขข้อมูลได้ และยังสามารถตรวจสอบว่าผู้สมัครทำการชำระเงินค่าลงทะเบียนเรียบร้อยแล้วหรือยัง
- ระบบสรุปข้อมูล (Information Summary) โดยระบบนี้จะทำการรวมข้อมูลทั้งหมดของงานประชุมไว้ที่ Dashboard ซึ่งมีหน้าสรุปข้อมูลที่สำคัญของงานประชุมทั้งหมดเอาไว้

3.4 การออกแบบฐานข้อมูล



รูป 3.8 แผนภาพ ER Diagram ของการออกแบบฐานข้อมูล

เนื่องจากระบบใช้ฐานข้อมูลที่เป็นเอกเทศซึ่งไม่ได้เชื่อมโยงกับฐานข้อมูลเดิมของงานประชุม ดังนั้นทางระบบจึงต้องร้องขอข้อมูลตามรูปแบบที่ได้ถูกกำหนดไว้เพื่อให้ผู้จัดงานสามารถนำข้อมูลเดิมมาใช้ได้ โดยใช้วิธีนำเข้าข้อมูล (import) ในรูปแบบไฟล์ CSV ได้

จากแผนภาพในรูปที่ 3.9 ซึ่งเป็น ER Diagram แสดงความสัมพันธ์ระหว่างแต่ละ Entities โดยจะแบ่งเป็น 4 Entities ได้แก่ Attendee, Organizer, Conference และ Payment โดยจะทำหน้าที่เก็บข้อมูลที่เกี่ยวข้องแต่การนำมาใช้งานจะมีการนำมาเข้าร่วมกัน (Join) แต่ละ Entity เพื่อนำ Attributes มาใช้ประกอบกัน

ฐานข้อมูลที่ระบบของเราใช้จะเป็นฐานข้อมูล SQL โดยใช้โปรแกรม MySQL Workbench ในการสร้างและจัดเก็บ โดยทางระบบจะแบ่งฐานข้อมูลออกเป็นส่วนดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลงานประชุม

ฐานข้อมูลส่วนนี้จะใช้Conference Id ในการแยกแต่ละงานประชุมออกจากกัน โดยประเภทที่เก็บข้อมูลจะเกี่ยวกับชื่องานประชุม , วันที่จัดงาน และราคาในการลงทะเบียนต่าง ๆ โดยมีโครงสร้างดังนี้

ตารางที่ 3.1 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลงานประชุม

Conference ID	Conference Name	Create By	Create Date	Start Date	Finish Date
---------------	-----------------	-----------	-------------	------------	-------------

Early Deadline	Payment Deadline	Early Regis	Member Early	Regular Late	Member Late
----------------	------------------	-------------	--------------	--------------	-------------

Student	Visitor	Ex Dinner	Addition Ticket	Active Conference
---------	---------	-----------	-----------------	-------------------

จากตารางจะเห็นข้อมูลดังนี้

ตารางที่ 3.2 คำอธิบายสำหรับส่วนเก็บข้อมูลงานประชุม

Attribute	Data Type	Description
Conference ID	varchar(45)	เก็บ ID งานประชุมที่สร้างID อะไร โดยเรียงตามลำดับการสร้าง
Conference Name	varchar(45)	เก็บชื่องานประชุมที่สร้างขึ้น
Create By	varchar(45)	เก็บ e-mail ของผู้จัดงานที่สร้างงานประชุมขึ้นมา
Start Date	datetime	เก็บค่าวันที่ที่เริ่มงานประชุม
Finish Date	datetime	เก็บค่าวันที่ที่จบงานประชุม
Early Deadline	datetime	เก็บค่าวันที่ที่สามารถลงทะเบียนแบบ ล่วงหน้าได้
Payment Deadline	datetime	เก็บค่าวันที่วันสุดท้ายของการจ่ายเงินเพื่อเข้าร่วมงานประชุม
Early Regis	int	เก็บราคาในการลงทะเบียนแบบ ล่วงหน้า
Member Early	int	เก็บราคาในการลงทะเบียนแบบ ล่วงหน้าและเป็นสมาชิก
Regular Late	int	เก็บราคาในการลงทะเบียนแบบ ค่าเช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Member Late	int	เก็บราคาในการลงทะเบียนแบบ ค่าเช่าและเป็นสมาชิก
Student	int	เก็บราคาในการลงทะเบียนแบบ นักเรียน
Visitor	int	เก็บราคาในการลงทะเบียนแบบ ผู้เข้าร่วมงานปกติ
Ex Dinner	int	เก็บราคาในการซื้อตัวอาหารเพิ่มเติม
Addition Ticket	int	เก็บราคาในการซื้อตัวอย่างอื่นเพิ่มเติม
Active Conference	varchar(45)	เก็บคำว่า งานประชุมนี้ยัง active อยู่หรือไม่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลผู้ใช้ระบบ

ฐานข้อมูลส่วนนี้จะใช้ User Id ในการแยกแต่ละผู้ใช้งานออกจากกัน โดยมีการเก็บข้อมูลทั้งหมดของผู้ใช้งานระบบเช่น ข้อมูลพื้นฐาน หรือข้อมูลใบเสร็จ ทั้งทางฝั่งของผู้เข้าร่วมงาน หรือผู้จัดงาน โดยมีโครงสร้างดังนี้

ตารางที่ 3.3 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ

User ID	Title	First Name	Middle Name	Last Name	Email	Password
---------	-------	------------	-------------	-----------	-------	----------

Affiliation	Address1	Address2	City
-------------	----------	----------	------

State	Country	Postcode	Phone Number	Fax	Invoice Name	Vat ID
-------	---------	----------	--------------	-----	--------------	--------

InvoiceAdd1	InvoiceAdd2	Invoice City	Invoice State	Invoice Country	Invoice ZIP
-------------	-------------	--------------	---------------	-----------------	-------------

จากตารางจะเห็นข้อมูลดังนี้

ตารางที่ 3.4 คำอธิบายสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ

Attribute	Data Type	Description
User ID	int	เก็บ ID ว่าผู้เข้าร่วมที่เข้าร่วมงานประชุมว่า ID อะไร โดยเรียงตามลำดับการสร้าง
Title	varchar(45)	เก็บว่าคำนำหน้าชื่อผู้เข้าร่วมงานเป็นอะไร เช่น Prof. , Mr. หรือ Ms. เป็นต้น
First Name	varchar(45)	เก็บชื่อจริงของผู้เข้าร่วมงานประชุม
Middle Name	varchar(45)	เก็บชื่อย่อของผู้เข้าร่วมงานประชุม
Last Name	varchar(45)	เก็บนามสกุลของผู้เข้าร่วมงานประชุม
Email	varchar(45)	เก็บ e-mail ของผู้เข้าร่วมงานประชุม
Password	varchar(45)	เก็บ password ของผู้เข้าร่วมงานประชุม
Affiliation	varchar(45)	เก็บสถาบันของผู้เข้าร่วมงานประชุม
Address1	varchar(45)	เก็บที่อยู่ของผู้เข้าร่วมงานประชุม(ที่ 1)
Address2	varchar(45)	เก็บที่อยู่ของผู้เข้าร่วมงานประชุม(ที่ 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

City	varchar(45)	เก็บเมืองที่อยู่อาศัยของผู้เข้าร่วมงานประชุม
State	varchar(45)	เก็บเมืองที่อยู่อาศัยของผู้เข้าร่วมงานประชุม
Country	varchar(45)	เก็บประเทศที่อยู่อาศัยของผู้เข้าร่วมงานประชุม
Postcode	varchar(45)	เก็บรหัสไปรษณีย์ของผู้เข้าร่วมงานประชุม
Phone Number	int	เก็บเบอร์โทรศัพท์ของผู้เข้าร่วมงานประชุม
Fax	int	เก็บเบอร์แฟกซ์ของผู้เข้าร่วมงานประชุม
Invoice Name	varchar(45)	เก็บชื่อของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
Vat ID	varchar(45)	เก็บ Vat ID ของผู้จ่ายเงิน
InvoiceAdd1	varchar(45)	เก็บที่อยู่ของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
InvoiceAdd2	varchar(45)	เก็บที่อยู่ของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
Invoice City	varchar(45)	เก็บเมืองที่อยู่ของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
Invoice State	varchar(45)	เก็บเมืองที่อยู่ของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
Invoice Country	varchar(45)	เก็บประเทศของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม
Invoice ZIP	varchar(45)	เก็บ ZIP code ของผู้จ่ายเงินสำหรับผู้เข้าร่วมงานประชุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 ฐานข้อมูลส่วนที่ใช้ในการเก็บข้อมูลผู้เข้าร่วมงานประชุม

ฐานข้อมูลส่วนนี้จะใช้ member Id ในการแยกแต่ละผู้เข้าร่วมออกจากกัน โดยมีการเก็บข้อมูลว่าลงทะเบียนงานประชุมใดไปบ้าง ลงทะเบียนในสถานะอะไร และลงทะเบียนล่าช้าหรือไม่ รวมถึงการเก็บข้อมูลจ่ายเงินของผู้เข้าร่วมอีกด้วย โดยมีโครงสร้างดังนี้

ตารางที่ 3.5 โครงสร้างข้อมูลสำหรับส่วนเก็บข้อมูลผู้เข้าร่วมงานประชุม

Member ID	email	Conference ID	Conference Name	status	Pay Method	Regis Time
ParticipationType	Amount Paper	Registration Type	image	Pay Date	Pay Time	
Pay Amount						

จากตารางจะเห็นข้อมูลดังนี้

ตารางที่ 3.6 คำอธิบายสำหรับส่วนเก็บข้อมูลผู้เข้าร่วมงานประชุม

Attribute	Data Type	Description
Member ID	int	เก็บ ID ของงานประชุมว่า ID ที่เข้าร่วม โดยอิงจากส่วนเก็บข้อมูลสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ
email	varchar(45)	เก็บ email ของผู้เข้าร่วมงานประชุม โดยอิงจากส่วนเก็บข้อมูลสำหรับส่วนเก็บข้อมูลผู้ใช้ระบบ
Conference ID	int	เก็บ conference id ของงานประชุมที่สมัคร
Conference Name	varchar(45)	เก็บ conference name ของงานประชุมที่สมัคร
status	varchar(45)	เก็บสถานะการจ่ายเงินของผู้เข้าร่วมงานประชุม
Pay Method	varchar(45)	เก็บช่องทางการจ่ายเงินของผู้เข้าร่วมงาน
Regis Time	datetime	เก็บเวลาที่สมัครงานประชุม
Participation Type	varchar(45)	เก็บตำแหน่งที่สมัคร เช่น ผู้เขียน หรือ ผู้ร่วมเขียน เป็นต้น
Amount Paper	int	เก็บจำนวน paper ที่ลงทะเบียนในงานประชุม
Registration Type	varchar(45)	เก็บประเภทการลงทะเบียน เช่น ล่าช้า หรือ ล่วงหน้า
image	varchar(45)	เก็บรูปใบเสร็จชำระเงินของแต่ละงานประชุม
Pay Date	datetime	เก็บวันที่ชำระเงินของแต่ละงานประชุม
Pay Time	datetime	เก็บเวลาที่ชำระเงินของแต่ละงานประชุม
Pay Amount	int	เก็บจำนวนที่ชำระเงินของแต่ละงานประชุม

3.5 Software Requirement Specification

ความต้องการของระบบที่ออกแบบไว้ จะแสดงในตารางดังต่อไปนี้

ตารางที่ 3.7 แสดงรายการความสามารถของระบบ

ID	Requirement	Priority
P-01	The system should display information about system on first page of website.	Must Have
P-02A	The system should display 'Partner Conference'.	Must Have
P-02B	The system should link to 'Partner Conference' website.	Must Have
P-03	The system should have login and sign-up link on first page.	Must Have
P-04A	The system should have support system for helping the user.	Should Have
P-05	The system should have registration system.	Must Have
P-06	The system should have sign in system.	Must Have
P-07	The system should have password reset system.	Should Have
P-08A	The system should have multi-user accounts types.	Must Have
P-08B	The system should support for change account type from permit account.	Must Have
P-09	The system should have user information system for organizer.	Must Have
P-10	The system should release organizer's ID by manual input.	Must Have
P-11	The system should have file upload system.	Must Have
P-12	The system should have payment status tracking system.	Must Have
P-13	The system should display notification for important information after login by pop-up.	Should Have
P-14A	The system should have payment system.	Must Have
P-14B	The system should accept bank transfer.	Must Have
P-14C	The system should have payment notification system to organizer when pay via bank transfer.	Should Have
P-14D	The system should accept pay for third person.	Must Have
P-14E	The system should make payment by scan QR Code.	Should Have
P-15	The system should have electronic signature system.	Should Have

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้ใช้ระบบจะขึ้นอยู่กับเงื่อนไขที่แนบมา

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P-16	The system should release invoice for customer.	Must Have
P-17	The system should release purchase order.	Should Have
P-18	The system should have cancel option for PO, receipt and release documents.	Must Have
P-19	The system should have specification conference's information.	Should Have
P-20	The system should release report to csv, pdf when user request.	Should Have



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

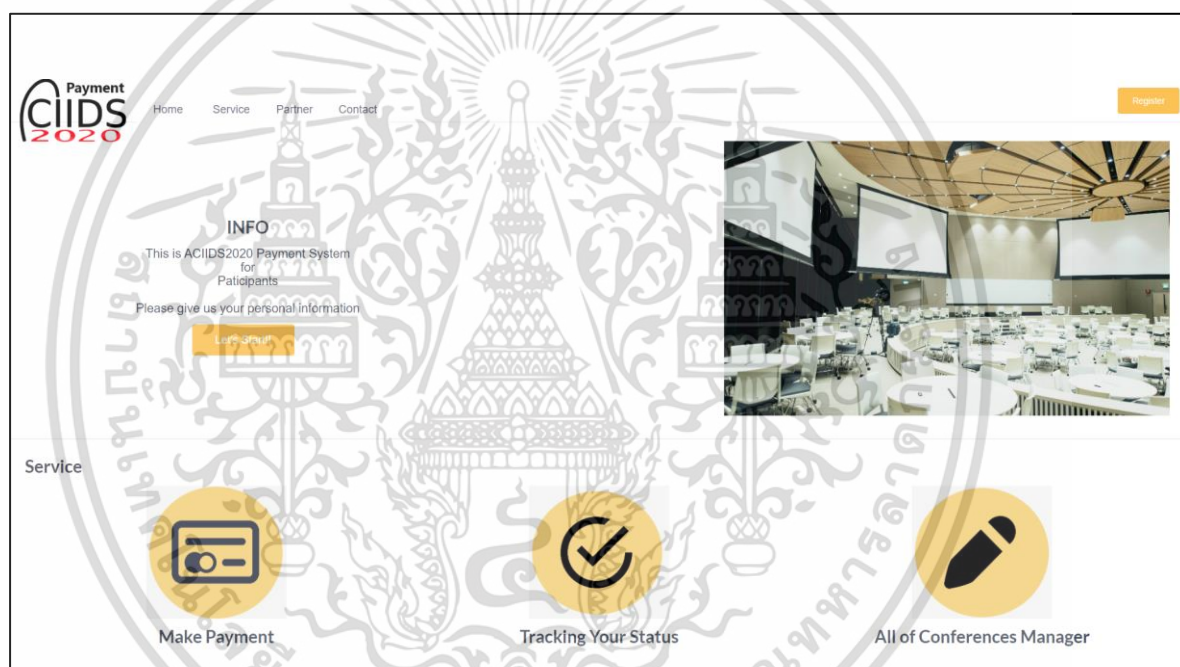
บทที่ 4

ผลการดำเนินงาน

ในส่วนนี้จะแสดงให้เห็นถึงผลการดำเนินงานทั้งหมดในช่วงเวลา 3 เดือน ของการพัฒนาและแก้ไขระบบต่าง ๆ ดังนี้

4.1 หน้าแรก

โดยทางผู้จัดทำได้ทำการแก้ไขและปรับปรุงให้หน้าแรกของระบบมีความเป็นระเบียบและเหมาะสมสำหรับงานประชุมทางด้านวิชาการมากขึ้น



รูป 4.1 หน้าแรกของระบบบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ระบบลงทะเบียนข้อมูลส่วนตัว

โดยทางผู้จัดทำได้สร้างระบบลงทะเบียนขึ้นมาเพื่อเก็บข้อมูลที่จำเป็นทั้งหมดสำหรับผู้เข้าร่วมงาน

Participant detail:

* Title

* Last name / Family name

* First name

Middle name

Affiliation

Address line 1

Address line 2

City

State

* Country

Postcode / ZIP code

* Phone

* Fax

* E-mail address

* Password

* Confirm Password

Invoice detail:

Name of the recipient of the invoice

Recipient's VAT ID

Please provide the invoice recipient's address

รูป 4.2 ระบบลงทะเบียนข้อมูลส่วนตัวบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ระบบ Login เข้าสู่หน้าหลักของผู้ใช้งาน

โดยระบบนี้จะเป็นระบบเพื่อคัดกรองว่า user เป็น ผู้เข้าร่วมงาน หรือ ผู้จัดงาน โดยถ้า user ลืมรหัสผ่าน สามารถเข้าไปสร้างใหม่ได้ ตาม Link ที่อยู่ใน website

รูป 4.3 ระบบ Login เข้าสู่หน้าหลักของผู้ใช้งาน

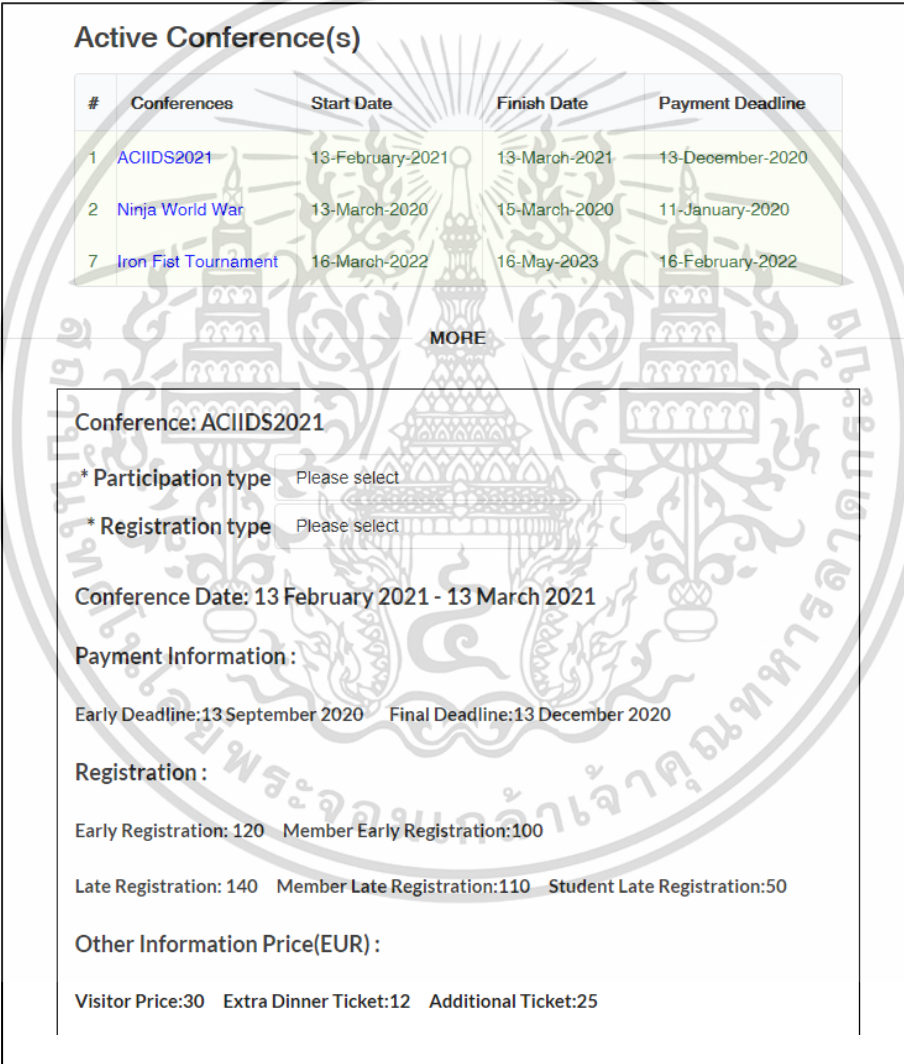
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ระบบสำหรับ Attendee

หลังจาก user ที่เป็นผู้เข้าร่วมงาน ทำการ login เข้าสู่ระบบ ระบบจะทำการแสดงหัวข้อ ต่าง ๆ ที่ผู้เข้าร่วมสามารถทำได้ ดังนี้

4.4.1 ระบบลงทะเบียนงานวิชาการ

การเลือกลงทะเบียนสำหรับงานวิชาการต่าง ๆ จะถูกเลือกภายในหน้านี้ โดยผู้เข้าร่วมจะสามารถเห็นข้อมูลทั้งหมดของงานประชุมวิชาการที่เลือกได้ เช่น วันที่เริ่มต้น , ค่าสมัครลงทะเบียน เป็นต้น



Active Conference(s)

#	Conferences	Start Date	Finish Date	Payment Deadline
1	ACIIDS2021	13-February-2021	13-March-2021	13-December-2020
2	Ninja World War	13-March-2020	15-March-2020	11-January-2020
7	Iron Fist Tournament	16-March-2022	16-May-2023	16-February-2022

MORE

Conference: ACIIDS2021

* Participation type

* Registration type

Conference Date: 13 February 2021 - 13 March 2021

Payment Information :

Early Deadline:13 September 2020 Final Deadline:13 December 2020

Registration :

Early Registration: 120 Member Early Registration:100

Late Registration: 140 Member Late Registration:110 Student Late Registration:50

Other Information Price(EUR) :

Visitor Price:30 Extra Dinner Ticket:12 Additional Ticket:25

รูป4.3 ระบบลงทะเบียนงานวิชาการบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 ระบบสรุปผลการลงทะเบียน

โดยเป็นระบบสำหรับสรุปผล ว่าผู้เข้าร่วมงานประชุมได้ลงทะเบียนในตำแหน่งใด และต้องจ่ายเท่าไร ซึ่งจะแสดงผลออกมาในรูปแบบของ invoice ที่สามารถ print ออกมาได้

Conference: ACIIDS2021

Invoice

KMIT Ladkrabang Co., Ltd. (on behalf of "The 12th Asian Conference on Intelligent Information and Database Systems (ACIIDS 2020)")
 1 Soi Chalongkrung 1, Ladkrabang,
 Bangkok, Thailand , 10520
 VAT ID: 010555155467
Recipient:Thawwin
Recipient VAT ID:11101010
Invoice #ACIIDS_KMITL_ID_2
Date:2020-03-03T14:31:12.000Z

No.	Description	Unit Cost (EUR)	Quantity	Price (EUR)
1	Registration fee for "ACIIDS2021" Registration ID: R0062 Participant type: "Author" Participant's name:thawwin	600	1	600
2	Registration type: "Member Late registration"	110	1	110
Extra:		Total		710
		Amount Paid		0.00
		Balance Due		710

Payment Terms

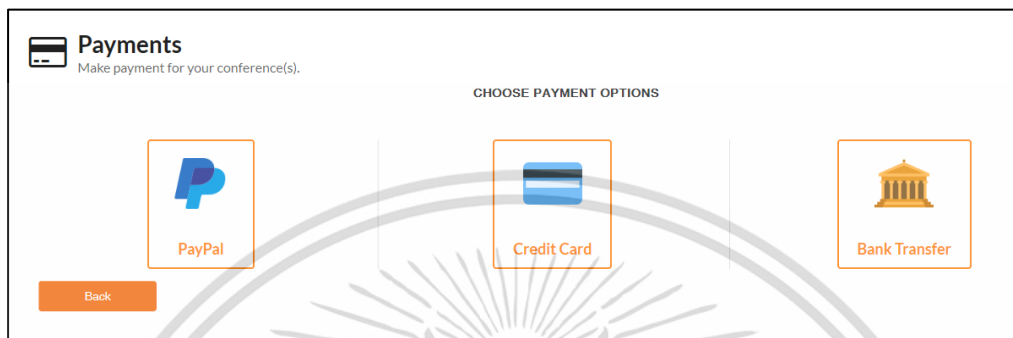
- Payment can be made by credit cards (online) or bank transfer.
- Account detail for bank transfer:
 Bank name: SIAM COMMERCIAL BANK PCL
 SWIFT Code: SICOTHBK
 Account no.: 414-050200-5 / Account name: KMIT LADKRABANG COMPANY LIMITED
 Branch code: 088 / Address: 3 MU2 CHALONGKRUNG RD., LADKRABANG
 City: BANGKOK / Country: THAILAND / Tel. [+66 2544 1000](tel:+6625441000)
 Reference/Message: ACIIDS R0062

รูป4.4 ระบบสรุปผลการลงทะเบียนบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.3 ระบบชำระเงินค่าลงทะเบียน

ระบบนี้จะให้ผู้เข้าร่วมชำระเงินค่าลงทะเบียนที่ตนเองต้องชำระ สามารถแบ่งวิธีการชำระเงินออกเป็น 3 ประเภท คือ PayPal , Credit Card และ โอนผ่านธนาคารแล้ว Upload สลิปเข้าไปในระบบ



รูป 4.5 ระบบชำระเงินค่าลงทะเบียน

4.4.4 ระบบแก้ไขข้อมูลสำหรับผู้เข้าร่วมงาน

ระบบนี้สามารถทำให้ผู้เข้าร่วมงานได้แก้ไขข้อมูลที่ผิดพลาดได้ เช่น ข้อมูลส่วนตัว หรือ ข้อมูลใบเสร็จ เป็นต้น

Participant detail:	
Title	Mr.
Last name / Family name	boonchoen
First name	thawin
Middle name	naruto
Affiliation	harvard
Address line 1	2132
Address line 2	asd
City	bangkok
State	bangkok
Country	Thailand
Postcode / ZIP code	10310
Phone	0945497733
Fax	0945497733

รูป 4.6 ระบบแก้ไขข้อมูลสำหรับผู้เข้าร่วมงาน ด้านข้อมูลส่วนตัว

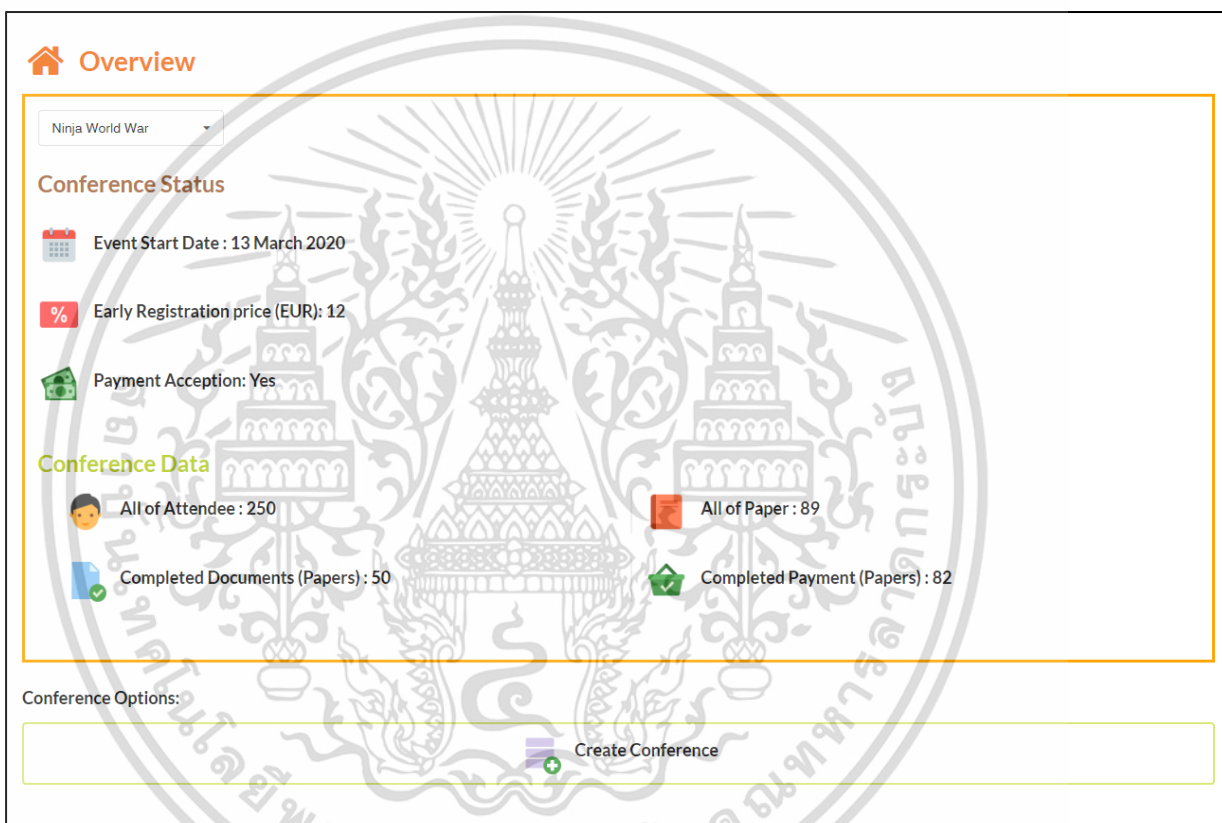
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ระบบสำหรับ Organizer

หลังจาก user ที่เป็นผู้จัดงาน ทำการ login เข้าสู่ระบบ ระบบจะทำการแสดงหัวข้อ ต่าง ๆ ที่ผู้จัดงานสามารถทำได้ ดังนี้

4.5.1 ระบบเพื่อดูภาพรวม

ระบบนี้มีไว้สำหรับผู้จัดงานดูข้อมูลคร่าวๆของงานประชุมต่าง ๆ ได้ โดยสามารถเข้าสู่ระบบสร้างงานประชุมผ่านระบบนี้ได้ด้วย



The screenshot displays the 'Overview' page for a conference titled 'Ninja World War'. The page is divided into several sections:

- Conference Status:**
 - Event Start Date: 13 March 2020
 - Early Registration price (EUR): 12
 - Payment Acceptance: Yes
- Conference Data:**
 - All of Attendee: 250
 - All of Paper: 89
 - Completed Documents (Papers): 50
 - Completed Payment (Papers): 82
- Conference Options:**
 - Create Conference

รูป 4.7 ระบบเพื่อดูภาพรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 ระบบสร้างงานประชุม

ผู้สร้างงานประชุมต่าง ๆ จะต้องเข้ามาในระบบนี้ เพื่อทำการสร้างงานประชุม โดยจำเป็นต้องใส่ข้อมูลต่าง ๆ ให้ครบถ้วน

Create a Conference

Conference Name:

Starting Date:

Finishing Date:

Early Birth Deadline:

Payment Deadline:

Registration Rate:

Early Birth Registration	Member Early Birth Registration	Regular Late Registration
EUR Amount <input type="text" value="0.00"/>	EUR Amount <input type="text" value="0.00"/>	EUR Amount <input type="text" value="0.00"/>
Member Late Registration	Student Late Registration	Visitor (Non-Paper)
EUR Amount <input type="text" value="0.00"/>	EUR Amount <input type="text" value="0.00"/>	EUR Amount <input type="text" value="0.00"/>

Extra Registration:

Exclusive Dinner:	1-Day Pass City Tour
EUR Amount <input type="text" value="0.00"/>	EUR Amount <input type="text" value="0.00"/>

รูป 4.8 ระบบสร้างงานประชุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.3 ระบบแก้ไขงานประชุม

ในระบบนี้ ผู้จัดงานสามารถเข้ามาแก้ไขข้อมูลงานประชุมได้ และสามารถเลือกได้ว่างานไหนจบไปแล้วบ้าง โดยการกดปุ่ม Active Check หรือ ลบงานประชมนั้น ๆ ได้

Conferences

Beware to config conference data.
It will effect for payment rate and attendee accounts.

ID	Conferences	Start Date	Finish Date	Payment Deadline	Active Check	Delete Conference
1	ACIIDS2021	13-February-2021	13-March-2021	13-December-2020	Enable	Delete
2	Ninja World War	13-March-2020	15-March-2020	11-January-2020	Enable	Delete
7	Iron Fiet Tournament	16-March-2022	16-May-2023	16-February-2022	Disable	Delete

Conference Options:

Create Conference

รูป4.9 ระบบแก้ไขงานประชุม หน้าเลือกงานประชุมที่ต้องการแก้ไข

4.5.4 ระบบตรวจสอบผู้เข้าร่วมงานประชุม

ระบบนี้จะอนุญาตให้ผู้จัดงานตรวจสอบข้อมูลรวมไปถึงการแก้ไขข้อมูลต่าง ๆ ของผู้เข้าร่วมงานในแต่ละงานประชุมได้ โดยระบบนี้สามารถเข้าถึงระบบยืนยันการชำระเงินของผู้เข้าร่วมงานได้

Attendee Data

Ninja World War

Attendee Name	Email	Phone Number	Payment Status	Delete User
Mr.thawin boonchoen	thawin9218@gmail.com	0946497733	In Procees	Delete
Prof.T K	kd@gmail.com	21212	In Procees	Delete

รูป4.10 ระบบตรวจสอบผู้เข้าร่วมงานประชุมบางส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.5 ระบบยืนยันการชำระเงินของผู้เข้าร่วมงาน

ระบบนี้จะทำการตรวจสอบว่าผู้เข้าร่วมงานได้ทำการ Upload สลิปโอนเงินเข้าสู่ระบบหรือไม่ ซึ่งถ้าผู้จัดงานทำการยืนยันการโอนเงิน payment status จะเปลี่ยนเป็น Complete

Ninja World War
▼

Payment Status: in Process

Payment Detail:

Slip Photo

Payment Date:	18/08/2561
Payment Time:	8.55
Amount:	5000

ยกเลิก
ต่อไป

Back
Confirm Payment

รูป4.11 ระบบยืนยันการชำระเงินของผู้เข้าร่วมงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ระบบที่ได้พัฒนาขึ้นมาเป็นระบบสำหรับการอำนวยความสะดวกให้ทั้งผู้เข้าร่วมงานและผู้จัดงาน ซึ่งจากการพัฒนาระบบในปัจจุบันนี้สามารถรองรับได้หลากหลายงานประชุม ระบบสามารถรับการชำระเงินได้ และเก็บสถานะพร้อมกับสลิปเงินเข้าสู่ฐานข้อมูลได้ สามารถแก้ไขข้อมูลผู้เข้าร่วมงานและข้อมูลของงานประชุมที่ต้องการได้ ซึ่งสามารถสรุปออกมาเป็นตารางได้ดังนี้

ตารางสรุปความสามารถของระบบ

ID	Requirement	Available
P-01	The system should display information about system on first page of website.	Yes
P-02A	The system should display 'Partner Conference'.	Yes
P-02B	The system should link to 'Partner Conference' website.	Yes
P-03	The system should have login and sign-up link on first page.	Yes
P-04A	The system should have support system for helping the user.	Yes
P-05	The system should have registration system.	Yes
P-06	The system should have sign in system.	Yes
P-07	The system should have password reset system.	Yes
P-08A	The system should have multi-user accounts types.	Yes
P-08B	The system should support for change account type from permit account.	Yes
P-09	The system should have user information system for organizer.	Yes
P-10	The system should release organizer's ID by manual input.	Yes
P-11	The system should have file upload system.	Yes
P-12	The system should have payment status tracking system.	Yes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P-13	The system should display notification for important information after login by pop-up.	No
P-14A	The system should have payment system.	Yes
P-14B	The system should accept bank transfer.	Yes
P-14C	The system should have payment notification system to organizer when pay via bank transfer.	No
P-14D	The system should accept pay for third person.	Yes
P-14E	The system should make payment by scan QR Code.	No
P-15	The system should have electronic signature system.	No
P-16	The system should release invoice for customer.	Yes
P-17	The system should release purchase order.	No
P-18	The system should have cancel option for PO, receipt and release documents.	No
P-19	The system should have specification conference's information.	Yes
P-20	The system should release report to csv, pdf when user request.	No

ฟังก์ชันการทำงานบางส่วนสามารถใช้งานได้จริง แต่บางฟังก์ชันการทำงานไม่สามารถทำงานได้จริงหากไม่ได้เชื่อมโยงเข้ากับฐานข้อมูลและต้องกำหนด Row ของฐานข้อมูลให้ตรงก่อนจึงจะสามารถทำงานได้

5.2 แนวทางการพัฒนาต่อ

- 1) พัฒนาให้ระบบสามารถรองรับการชำระเงินและแสดงผลหลายสกุลเงินได้
- 2) พัฒนาฟังก์ชันแต่ละส่วนเพิ่มเติมให้สามารถทำงานได้ดีและมีประสิทธิภาพมากขึ้น
- 3) พัฒนาระบบความปลอดภัยในเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. เครื่องมือที่ใช้ในการพัฒนาระบบ

1.1 Frameworks ในส่วนของ Front-End

- **Semantic Ui React** ใช้สำหรับการสร้าง Component ต่าง ๆ ที่เป็นส่วนของ User Interface เช่น ปุ่มกด, ตาราง เป็นต้น



รูป 1ก. สัญลักษณ์ของ Semantic Ui React

- **Styled-Component** ใช้ในการกำหนด Custom Component เองได้ ทำให้เราสามารถใส่ Tagที่กำหนดขึ้นมาเองสำหรับการออกแบบโดยการเขียนจะใช้ CSSในการเขียนปกติ



รูป 2ก. สัญลักษณ์ของ Styled-Component

- **Material-Ui** เป็นframework ที่ช่วยในการออกแบบGUI ต่าง ๆ คล้ายกับ Semantic Ui แต่เป็นของบริษัท Google

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3ก. สัญลักษณ์ของ Material-Ui

1.2 Frameworks ในส่วนของ Back-End

ส่วนระบบหลังบ้าน (Back-End) จะใช้ Node.js ในเชื่อมต่อ โดยเรียกใช้งาน Node Package Manager ดังนี้

- npm



รูป 4ก. สัญลักษณ์ของ npm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Yarn



รูป 5ก. สัญลักษณ์ของ yarn

- **Swagger** ใช้ในการจัดการเพื่อให้สามารถใช้ทดสอบ API ที่เขียนขึ้นมาได้ว่าสามารถใช้งานได้หรือไม่โดยที่ยังไม่ Implement เข้ากับระบบจริง

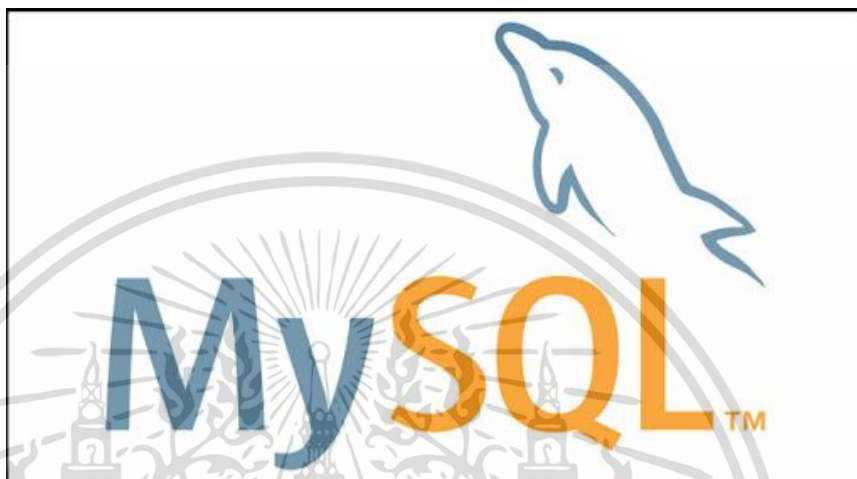


รูป 6ก. สัญลักษณ์ของ Swagger

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ฐานข้อมูล

โครงการนี้ใช้ฐานข้อมูลแบบ SQL โดยใช้โปรแกรม MySQL Workbench ซึ่งเป็นฐานข้อมูลที่สามารถใช้งานได้โดยไม่เสียค่าใช้จ่าย และสามารถจำลองเป็นเครื่อง Sever แบบ Localhost ซึ่งสามารถกำหนดเป็น Connection ในการเชื่อมต่อได้



รูป 7ก. สัญลักษณ์ของ My SQL

1.4 การจัดการโปรเจก

ในการจัดการไฟล์โปรเจกจะใช้ Github เข้ามาช่วยในการทำงานร่วมกัน ซึ่งจะใช้งานผ่าน GitKraken



รูป 8ก. สัญลักษณ์ของ GitKraken

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 IDE

ทางผู้จัดทำได้ตกลงที่จะใช้ IDE ของ Microsoft ที่ชื่อว่า Visual Studio Code เนื่องจากสามารถใช้งานได้โดยไม่เสียค่าใช้จ่ายและสามารถลงโปรแกรมเสริม(Extension)ได้



รูป 9ก. สัญลักษณ์ของ Visual Studio Code

และได้ลงโปรแกรมจัดการในการเขียนโค้ดที่ชื่อว่า ESLint ซึ่งจะทำหน้าที่ตรวจสอบการเขียนโค้ดให้ถูกต้อง และการจัดระเบียบการเขียนสำหรับภาษา JavaScript



รูป 10ก. สัญลักษณ์ของ ESLint

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ขั้นตอนการติดตั้ง

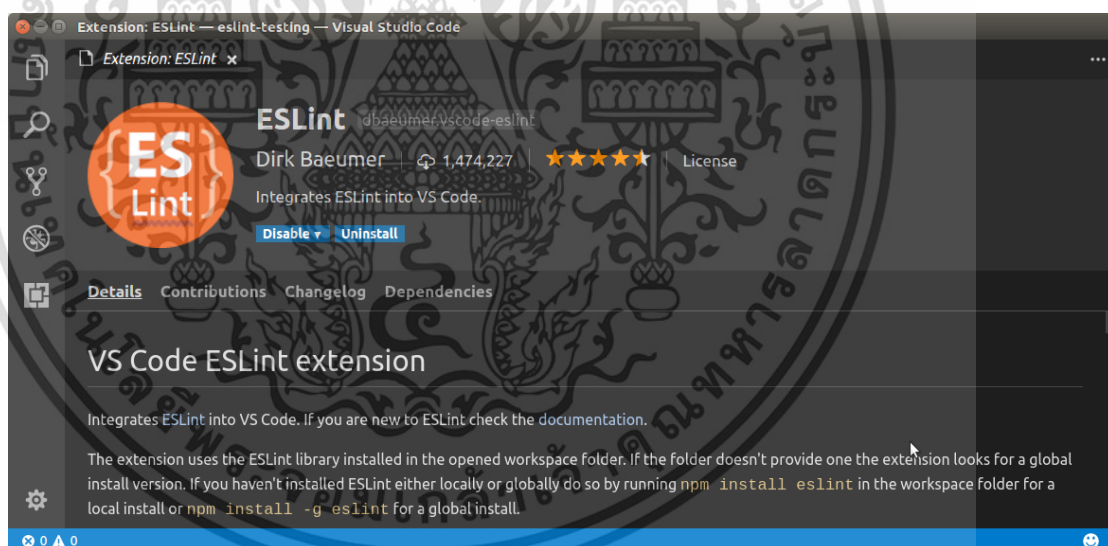
ก่อนการเริ่มพัฒนาระบบ จำเป็นต้องติดตั้งเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนาระบบ โดยมีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 ติดตั้ง Node.js และ Node Package Manager ซึ่งตัว Node Package Manager สามารถใช้ npm หรือ yarn ก็ได้ โดนสามารถใส่คำสั่งในการติดตั้งดังต่อไปนี้

```
$ npm install
```

```
$ yarn install
```

ขั้นตอนที่ 2 ติดตั้ง IDE ในการพัฒนา ซึ่งแนะนำให้ใช้โปรแกรม Visual Studio Code ที่สามารถติดตั้งส่วนขยาย(Extension) ชื่อว่า ESLint ทำหน้าที่ตรวจสอบการเขียนโค้ดให้ถูกต้อง และการจัดระเบียบการเขียนสำหรับภาษา JavaScript



รูป 11ก. หน้าจอโปรแกรม VS Code ขณะทำการติดตั้ง ESLint

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 ติดตั้งโปรแกรม GitKraken พร้อมสมัครบัญชีผู้ใช้ จากนั้นทำการ clone ไฟล์โปรเจกต์จาก Github เพื่อนำมาพัฒนาต่อ

```
pip install git
git clone https://github.com/{username}/{filename}.git
```

ขั้นตอนที่ 4 ทำการติดตั้งไฟล์ Module ของ React Framework ที่ไฟล์โปรเจกต์ทำการ clone มาจาก Github โดยใช้คำสั่งดังนี้

```
yarn init
yarn add react react-dom
```

ขั้นตอนที่ 5 ทำการติดตั้ง Package ต่าง ๆ ที่ใช้ในการพัฒนาระบบ ได้แก่ semantics-ui, react-router, material-ui, styled component และ grid foundation

ขั้นตอนที่ 6 ทำการติดตั้ง ฐานข้อมูลโดยใช้โปรแกรม MySQL Workbench และทำการสร้างฐานข้อมูล

```
pip install mysql-server
mysql_secure_installation
```

ขั้นตอนที่ 7 ทำการติดตั้ง nvm ซึ่งเป็นตัวจัดการ Version ของ node.js

ขั้นตอนที่ 8 เลือกใช้ nodejs version 6.14.1 โดยใช้คำสั่ง

```
nvm install 6.14.1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 9 ให้ทำการทดสอบระบบโดยทดลองสั่งเริ่มต้น โปรเจกต์ ซึ่งสามารถใช้คำสั่งในการรัน โปรเจกต์ได้ดังนี้

yarn start ในฝั่ง Front-End และ nodemon server ในฝั่ง Back-End



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์ในระบบของเราจะแบ่งออกเป็น 2 ส่วนหลักคือ

- **Frontend** เป็นไฟล์ส่วนของ Front-End ซึ่งเกี่ยวข้องกับการแสดงผลออกเป็นหน้าเว็บไซต์ หรือ GUI และการเปลี่ยนหน้าเว็บ การจัดการข้อมูล State ต่างๆ และการส่งข้อมูลและไฟล์หน้าเว็บไซต์ ไปยังฐานข้อมูล

- **Backend** เป็นไฟล์ส่วนของ Back-End ซึ่งเกี่ยวข้องกับการรับ-ส่งข้อมูลจากหน้าบ้านไปยังฐานข้อมูล และ API (Application Programming Interface) ต่างๆ ที่จะเป็นตัวกลางที่คอยจัดการข้อมูลในระบบให้สามารถแก้ไขหรือให้ข้อมูลได้อย่างถูกต้อง

1. Frontend

Name	Date modified	Type	Size
.vscode	2/4/2563 4:22	File folder	
assets	7/11/2562 15:26	File folder	
node_modules	2/3/2563 3:14	File folder	
src	24/10/2562 10:58	File folder	
webpack	24/10/2562 10:58	File folder	
.eslintrc	24/10/2562 10:58	ESLINTRC File	1 KB
.gitignore	24/10/2562 10:58	Text Document	1 KB
.travis.yml	24/10/2562 10:58	YML File	1 KB
package.json	2/3/2563 3:04	JSON File	3 KB
package-lock.json	31/10/2562 16:07	JSON File	441 KB
README.md	1/11/2562 15:24	MD File	1 KB
yarn.lock	2/3/2563 3:04	LOCK File	319 KB

รูป 1x. โครงสร้างของข้อมูลของโฟลเดอร์ Frontend

โดยโฟลเดอร์ต่างๆสามารถอธิบายได้ดังนี้

- **.vscode** เป็นที่เก็บไฟล์ setting ของตัวโปรแกรม Visual Studio Code สำหรับ Project Frontend นี้

- **assets** เป็นโฟลเดอร์ที่เก็บไฟล์รูปภาพหรือ Icon ที่ใช้แสดงผลในเว็บไซต์ที่เราสร้างขึ้นมา โดยข้างในโฟลเดอร์ก็จะแยกเป็นสถานที่ของเว็บไซต์ที่ต้องการใช้

Name	Date modified	Type	Size
Header	24/10/2562 10:58	File folder	
Home	7/11/2562 15:26	File folder	
Login	7/11/2562 15:26	File folder	
Organizer	24/10/2562 10:58	File folder	
Payments	7/11/2562 15:26	File folder	
Setting	24/10/2562 10:58	File folder	
ACIIDS.png	7/11/2562 15:26	PNG File	15 KB
ACIIDS2.png	7/11/2562 15:26	PNG File	17 KB
FullLogo.png	24/10/2562 10:58	PNG File	17 KB
manifest.json	24/10/2562 10:58	JSON File	1 KB
paymeSYSlogo.png	24/10/2562 10:58	PNG File	1 KB
profilepic.png	24/10/2562 10:58	PNG File	6 KB
styled.js	24/10/2562 10:58	JavaScript File	2 KB

รูป 2ข. ไฟล์บางส่วนภายในโฟลเดอร์ assets

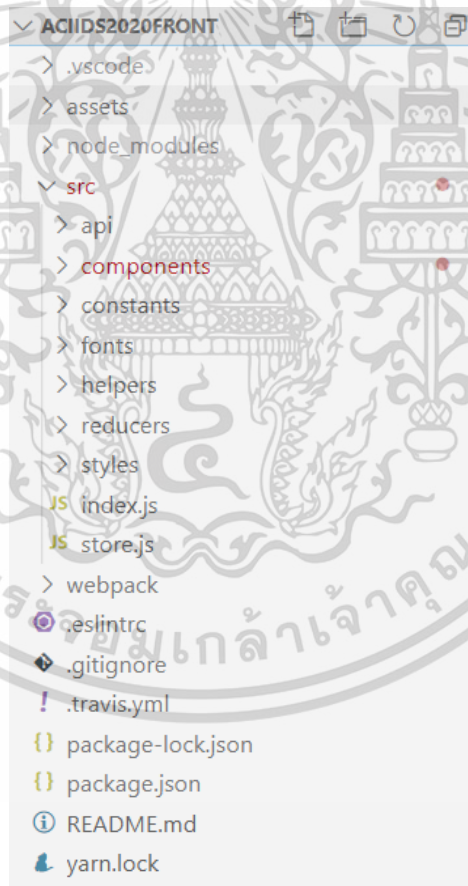
- **node_modules** เป็นโฟลเดอร์ที่เก็บข้อมูล Package และ เครื่องมือสำคัญต่าง ที่ทำให้ตัว Project Frontend นี้ สามารถรันใช้งานได้ โดย ข้อมูลในโฟลเดอร์นี้ห้ามถูกแก้ไขเด็ดขาดหรือลบออกอย่างผิดวิธีโดยเด็ดขาด
- **src** เป็นโฟลเดอร์ที่เก็บตัว Source Code ของแต่ละหน้าเว็บไซต์เอาไว้ทั้งหมดซึ่งจะอธิบายโดยละเอียดในตอนท้าย
- **webpack** เป็นที่เก็บไฟล์ webpack ซึ่งติดต่อสื่อสารกับโปรแกรม Web Browser และแปลงภาษาที่ทำให้ Web Browser เข้าใจ ซึ่งได้กล่าวเอาไว้ในบทที่ 2
- **.eslintrc** เป็นไฟล์ที่เอาไว้แก้ไขการตั้งค่าของโปรแกรมส่วนเสริมของ ESLint ซึ่งจะเป็นตัวที่ทำให้ Source Code มีความเป็นระเบียบละอ่านง่ายขึ้น
- **.gitignore** เป็นไฟล์ที่เอาไว้ระบุ โฟลเดอร์ที่ไม่ต้องการ Upload ลงไปใน git ด้วย
- **Package.json** เป็นไฟล์ที่คอยเก็บรายชื่อของเครื่องมือที่เราโหลดเข้ามาใช้งานใน Project นี้
- **Yarn.lock** เป็นไฟล์ที่คอยเก็บรายชื่อและ Version ที่เราทำการโหลดมาผ่านตัว Package Management ของ Yarn

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
body-parser@1.18.2:
  version "1.18.2"
  resolved "https://registry.yarnpkg.com/body-parser/-/body-parser-1.18.2.tgz#87678a19d84b47d859b83199bd59bce222b10454"
  dependencies:
    bytes "3.0.0"
    content-type "~1.0.4"
    debug "2.6.9"
    depd "~1.1.1"
    http-errors "~1.6.2"
    iconv-lite "0.4.19"
    on-finished "~2.3.0"
    qs "6.5.1"
    raw-body "2.3.2"
    type-is "~1.6.15"
```

รูป 3ข. ตัวอย่างข้อมูลที่เก็บอยู่ใน Yarn.lock

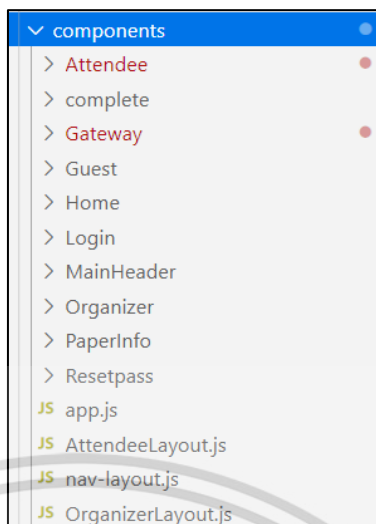
ไฟล์ที่อยู่ในโฟลเดอร์ src เป็นสิ่งที่มีความสำคัญมากเพราะเป็นส่วนของการแสดงผลขึ้นหน้าเว็บไซต์ รายละเอียดต่างๆของตัวเว็บไซต์ และไฟล์สำคัญที่ตัว React เรียกใช้งาน ซึ่งภายในโฟลเดอร์ดังกล่าวมีข้อมูลดังนี้



รูปที่ 4ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src

- **src/components** สำหรับเก็บไฟล์ Source Code ของการแสดงผลทั้งหมด โดยจะแบ่งเป็นส่วนๆ โดยโฟลเดอร์ย่อยนั้นคือแต่ละหน้าของเว็บไซต์ ซึ่งแต่ละหน้ามีดังนี้

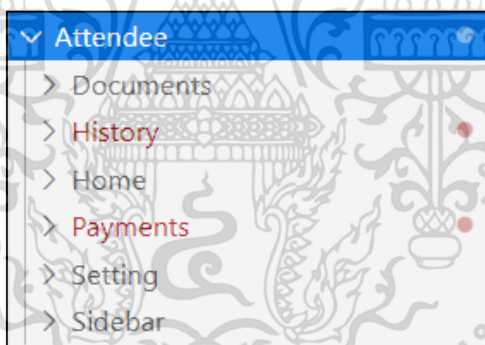
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components

โดยใน Components จะมีโฟลเดอร์จะจำแนกเป็น 3 ส่วนคือ

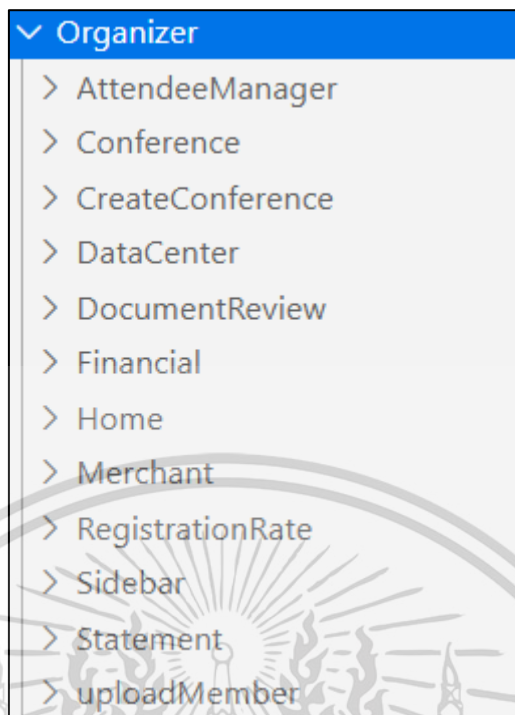
- หน้าเว็บไซต์ของ **Attendee** จะเป็นส่วนที่อยู่ใน Folder Attendee ซึ่งจะเข้าได้นั้นต้องมี Role เป็น Attendee



รูปที่ 6ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components/Attendee

- หน้าเว็บไซต์ของ **Organizer** จะเป็นส่วนที่อยู่ใน Folder Organizer ซึ่งจะเข้าได้นั้น ต้องมี Role เป็น Organizer เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7ข. โครงสร้างข้อมูลของโฟลเดอร์ frontend/src/components/Organizer

- เว็บไซต์ส่วนกลาง จะเป็นส่วนของ Folder ที่ไม่อยู่ทั้งใน Attendee และ Organizer ซึ่งจะเป็นส่วนของหน้ารับรอง หรือ หน้า Login
- นอกจากไฟล์ Source code ที่ไว้แสดงผลบนเว็บไซต์แล้ว ก็มีไฟล์ต่างๆที่เอาไว้คอยควบคุมเว็บไซต์ให้มีการทำงานที่ถูกต้องอีกด้วย ได้แก่
 - **App.js** เป็นตัวกลางที่ไว้เชื่อมต่อหน้าของทุกๆ หน้า ให้ทำการเข้าถึงกันได้ผ่าน Function <Route> ของ React-Router
 - **AttendeeLayout.js** เป็นส่วน Layout ของ Attendee เมื่อมีการ Login แล้ว Role ของผู้ใช้งานเป็น Attendee
 - **OrganizerLayout.js** เป็นส่วน Layout ของ Organizer เมื่อมีการ Login แล้ว Role ของผู้ใช้งานเป็น Organizer
 - **Nav-layout** เป็นส่วนของ Navbar (แถบ Menu ที่อยู่ด้านบน) ที่จะแสดงตอนไม่มีการ Login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

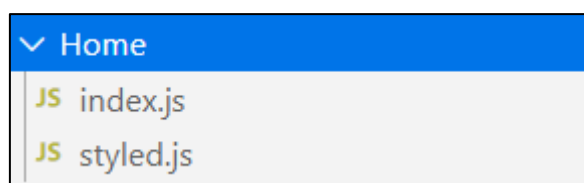
2. การเขียนโปรแกรม

ในการเขียนโปรแกรมนั้น เราจะมองแยกแต่ละส่วนเป็น Components ซึ่งแต่ละส่วนจะถูกเก็บแยกกัน ซึ่งแต่ละหน้าจะมีอยู่ 2 ไฟล์ คือ index.js และ styled.js ซึ่งแต่ละ Components จะมีการ exports จากนั้นจะทำการเรียกใช้ผ่าน tag ที่ถูก exports มา ยกตัวอย่างเช่น <OrganizerLayout> หรือ <AttendeeLayout> โดยที่มี <div> ครอบ

```
render() {
  let route;
  if (this.props.role === 'attendee') {
    route = (
      <NavLayout>
        <Switch>
          <Route path="/login" component={Login} />
          <Route path="/home" component={Home} />
          <Route path="/resetpass" component={Resetpass} />
          <AttendeeLayout>
            <Switch>
              <Route path="/attendee/home" component={AttendeeHome} />
              <Route path="/attendee/documents" component={Documents} />
              <Route path="/attendee/payments" component={Payments} />
              <Route path="/attendee/paymentssummary" component={PaymentSummary} />
              <Route path="/attendee/history" component={Histories} />
              <Route path="/attendee/setting" component={Setting} />
              <Route path="/uploadslip" component={uploadslip} />
              <Route path="/summary" component={Summary} />
              <Route path="/attendee/complete" component={complete} />
              <Redirect to={{ pathname: '/attendee/home' }} />
            </Switch>
          </AttendeeLayout>
        </Switch>
      </NavLayout>
    );
```

รูปที่ 7ข. ตัวอย่างโค้ด app.js (Frontemd)

โดยใน 1 Components นั้นจะมี อยู่ 2 ไฟล์คือ index.js หรือ ส่วนของ Source code และ Styled.js หรือส่วนของ CSS ที่ใช้สำหรับจัดตำแหน่งหรือ ตกแต่งหน้าเว็บไซต์สำหรับการใช้งาน ซึ่งจะถูกริเรียกใช้จากตัว index.js ผ่าน styled



รูปที่ 8ข. รูปตัวอย่างข้อมูลของ 1 Components

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1  import React, { Component } from "react";
2  import PropTypes from "prop-types";
3  import { connect } from "react-redux";
4  import { Redirect, Link } from "react-router-dom";
5  import { Form, Divider, Button, Icon } from "semantic-ui-react";
6  import { LoginBlock, Logo, FormBox, ButtonBox } from "../styled";
7  import logoPic from "../../assets/Login/ACIIDS2.png";
8  import ICSEC from "../../assets/Login/ICSEC.png";
9  import ICANN from "../../assets/Login/ICANN.png";
10 import Con1 from "../../assets/Login/Con1.png";
11
12 import { Login } from "../../api/login";
13
14 class LoginSentEmail extends Component {
15   static propTypes = {
16     userRole: PropTypes.string.isRequired,
17     // LoginSentEmail: PropTypes.func.isRequired,
18     // LoginSendOTP: PropTypes.func.isRequired,
19     match: PropTypes.string.isRequired
20   };
21
22   constructor(match) {
23     super(match);
24     this.state = {
25       status: 0, // Status of next button
26       email: "",
27       password: "",
28       Pic: ""
29     };
30   }
31
32   handleChange = (e, { name, value }) => {
33     console.log("SetState :", value);
34     this.setState({ [name]: value });
35   };

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

37 ✓ handleSubmit = e => {
38   e.preventDefault();
39   this.props.LoginSendEmail(this.state.email);
40   this.setState({ status: 1 });
41 };
42
43 ✓ handleSubmitLogin = e => {
44   e.preventDefault();
45   Login(this.state.email, this.state.password);
46   // window.location.reload();
47 };
48
49 ✓ RenderConPic = () => {
50   const paramCon = this.props.match.params.confer;
51   if (paramCon === "icesc") {
52     return <Logo src={ICSEC} />;
53   } else if (paramCon === "icann") {
54     return <Logo src={ICANN} />;
55   } else if (paramCon === "con1") {
56     return <Logo src={Con1} />;
57   }
58   return <div />;
59 };
60 LoginForm = () => {};
61 ✓ RenderPage = () => {
62   return (
63     <div>
64       <LoginBlock>
65         {this.RenderConPic()}
66         <Logo src={logoPic} />
67         <FormBox>
68           <Form onSubmit={this.handleSubmitLogin}>
69             <Form.Field>
70               <h3>Email Address</h3>
71               <Form.Input
72                 placeholder="Email (ex. 'aciids01@kmitl.ac.th')"
73                 name="email"
74                 type="email"
75                 onChange={this.handleChange}
76                 required
77               />
78             </Form.Field>
79             <Form.Field>
80               { /* <h3>Your email : {this.state.email}</h3> */ }
81               <h3>Password</h3>
82               <Form.Input
83                 placeholder="password"
84                 name="password"
85                 type="password"
86                 onChange={this.handleChange}
87                 required
88               />

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

89 </Form.Field>
90 <Link to="/resetpass">
91   <p align="left">Forgot Password?</p>
92 </Link>
93 <ButtonBox>
94   <Button
95     animated
96     color="orange"
97     onClick={this.handleSetStatus}
98     style={{ width: "150px" }}
99   >
100   <Button.Content visible>Login</Button.Content>
101   <Button.Content hidden>
102     <Icon name="right arrow" />
103   </Button.Content>
104 </Button>
105 <p />
106 <Link to="/guest" align="left">
107   <Button animated color="blue" style={{ width: "150px" }}>
108     <Button.Content visible>Create Account</Button.Content>
109     <Button.Content hidden>
110       <Icon name="address card icon" />
111     </Button.Content>
112   </Button>
113 </Link>
114 </ButtonBox>
115 </Form>
116 </FormBox>
117 </LoginBlock>
118 </div>
119 );
120 };
121

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

122 ✓ render() {
123 ✓   if (this.props.userRole === "attendee") {
124     return <Redirect to={{ pathname: "/attendee/home" }} />;
125   }
126   // if (this.props.userRole === 'attendee') {
127   //   return <Redirect to={{ pathname: '/attendee/home' }} />;
128   // }
129 ✓   else if (this.props.userRole === "organizer") {
130     return <Redirect to={{ pathname: "/organizer/home" }} />;
131   }
132
133   return <div>{this.RenderPage()}</div>;
134 }
135 }
136
137 ✓ const mapStateToProps = state => ({
138   userRole: state.loginOTPREducer.role
139 });
140
141 ✓ const mapDispatchToProps = dispatch => ({
142   // LoginSendEmail: email => LoginSendEmail(email)
143   // LoginSendOTP: (otp, email) => dispatch(LoginSendOTP(otp, email))
144 });
146 export default connect(mapStateToProps, mapDispatchToProps)(LoginSentEmail);

```

รูปที่ 10ข. รูปตัวอย่างโค้ดของ index.js (Login)

ข้างต้นเป็นไฟล์หน้าเว็บไซต์ที่ถูกเขียนอยู่ในไฟล์ index.js ของ Login ซึ่งเป็นหน้าไว้ใช้สำหรับ ล็อกอินเข้าไปใช้งานเว็บไซต์ โดยส่วนของโค้ด จะแบ่งเป็นส่วนๆ ดังนี้

```

1 import React, { Component } from "react";
2 import PropTypes from "prop-types";
3 import { connect } from "react-redux";
4 import { Redirect, Link } from "react-router-dom";
5 import { Form, Divider, Button, Icon } from "semantic-ui-react";
6 import { LoginBlock, Logo, FormBox, ButtonBox } from "../styled";
7 import logoPic from "../../assets/Login/ACTIDS2.png";
8 import ICSEC from "../../assets/Login/ICSEC.png";
9 import ICANN from "../../assets/Login/ICANN.png";
0 import Con1 from "../../assets/Login/Con1.png";
1
2 import { Login } from "../../api/login";
3

```

ส่วนที่ 1 เป็นส่วนของการ Import Package หรือ ข้อมูลต่างๆ ที่เราต้องการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

class LoginSentEmail extends Component {
  static propTypes = {
    userRole: PropTypes.string.isRequired,
    // LoginSentEmail: PropTypes.func.isRequired,
    // LoginSentOTP: PropTypes.func.isRequired,
    match: PropTypes.string.isRequired
  };

  constructor(match) {
    super(match);
    this.state = {
      status: 0, // Status of next button
      email: "",
      password: "",
      Pic: ""
    };
  }
}

```

ส่วนที่ 2 เป็นส่วนของการประกาศใช้ class name ของ Components และจะมีการประกาศ Constructor และ ค่า State ต่างๆ ที่ต้องใช้ภายใน class

```

handleSubmitLogin = e => {
  e.preventDefault();
  Login(this.state.email, this.state.password);
  // window.location.reload();
};

RenderConPic = () => {
  const paramCon = this.props.match.params.confer;
  if (paramCon === "icesc") {
    return <Logo src={ICSEC} />;
  } else if (paramCon === "icann") {
    return <Logo src={ICANN} />;
  } else if (paramCon === "con1") {
    return <Logo src={Con1} />;
  }
  return <div />;
}

```

ส่วนที่ 3 คือส่วนของ Function ต่างๆ ที่เราจะประกาศใช้ภายใน class นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

render() {
  if (this.props.userRole === "attendee") {
    | return <Redirect to={{ pathname: "/attendee/home" }} />;
  }
  // if (this.props.userRole === 'attendee') {
  // | return <Redirect to={{ pathname: '/attendee/home' }} />;
  // }
  else if (this.props.userRole === "organizer") {
    | return <Redirect to={{ pathname: "/organizer/home" }} />;
  }

  return <div>{this.RenderPage()}</div>;
}

```

ส่วนที่ 4 จะเป็นส่วนสำหรับการ Render โดยส่วนนี้จะเป็นส่วนของการประกาศให้แสดงผลออกไปยัง Web browser

```
export default connect(mapStateToProps, mapDispatchToProps)(LoginSentEmail);
```

ส่วนที่ 5 เป็นส่วนของการ Export ซึ่ง จะทำให้ exports Class นั้น ไปใช้ในที่อื่นได้ ถ้าทำการเรียกใช้ ยกตัวอย่างเช่น เมื่อทำการเขียน Components และ exports เสร็จ จะถูกนำไปเรียกใช้ที่ app.js ส่วนวิธีการ Export คือ export default (...ชื่อ class)

ส่วนการกำหนด style หรือการเขียน CSS นั้นจะเขียนอยู่อีกไฟล์หนึ่ง ที่ชื่อว่า styled.js โดยจะมีการเรียกใช้ styled component เพื่อที่ให้เราสามารถเขียน Tag แบบ Custom เองได้ เพื่อปรับใช้ในสถานะการที่แตกต่างกัน และสามารถหาแก้ไขได้ง่ายในภายภาคหน้า

```

import styled from 'styled-components';
import { Image } from 'semantic-ui-react';

export const LoginBlock = styled.div`
padding: 3%;
height: 1000px;
background: white;
`;

export const FormBox = styled.div`
padding-top: 1rem;
height: auto;
width: 450px;
display: block;
margin: auto;
`;

export const ButtonBox = styled.div`
padding-top: 1rem;
width: 150px;
display: block;
margin: auto;
`;

export const Logo = styled(Image)`
&&{
height: 200px;
width: 400px;
display: block;
margin-left: auto;
margin-right: auto;
}
`;

```

รูปที่ 11 ข. รูปตัวอย่างโค้ดของ styled.js (Login)

โดยการเขียน Tag แบบ Custom นั้นจะถูกเขียนโดยใช้ CSS ปกติ แต่จะมีบาง Syntax ที่พิเศษซึ่งจะถูกกำหนดไว้โดย Style components และจะต้องทำการ export ชื่อของ Components นั้นๆ มาก่อน แล้วนำไป import ใช้ในไฟล์ที่เราต้องการ

ในบาง Components อาจจะมี Components ย่อยภายในอีก ซึ่งจะเป็น Folder ย่อยภายใน แต่เมื่อทำการเปิดดูก็จะพบว่ารูปแบบการเขียน Components นั้นคล้ายคลึงกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.Backend

Name	Date modified	Type	Size
api	26/3/2563 3:45	File folder	
node_modules	18/4/2563 5:32	File folder	
public	19/4/2563 16:32	File folder	
.gitignore	24/10/2562 10:59	Text Document	1 KB
app.js	20/4/2563 0:15	JavaScript File	2 KB
eaodumpdatabase.sql	6/11/2562 19:36	SQL Text File	25 KB
favicon.ico	3/2/2563 23:16	Icon	0 KB
package.json	18/4/2563 5:32	JSON File	1 KB
package-lock.json	18/4/2563 5:32	JSON File	82 KB
server.js	3/2/2563 23:17	JavaScript File	1 KB

รูปที่ 12 ข. โครงสร้างข้อมูลของโฟลเดอร์ Backend

โดยโฟลเดอร์ต่างๆ สามารถอธิบายได้ดังนี้

- **api** คือโฟลเดอร์ที่เก็บรวบรวม API ไว้สำหรับรองรับการเรียกใช้งาน โดยข้างใน Folder นี้จะมีไฟล์ที่มีการเขียน API และจะแยกเป็นส่วนๆ ตามการใช้งาน

Name	Date modified	Type	Size
createTable.js	19/4/2563 16:12	JavaScript File	5 KB
memberConference.js	19/4/2563 17:19	JavaScript File	4 KB
paypal.js	19/4/2563 18:29	JavaScript File	8 KB
user.js	19/4/2563 17:49	JavaScript File	26 KB

รูปที่ 13 ข. โครงสร้างข้อมูลของโฟลเดอร์ Backend/api

ในส่วนของ Source code ใน Folder api จะเห็นได้ว่า จะแยกเป็นส่วนๆ ที่มีการใช้งานที่แตกต่างกัน หรือ มีหมวดหมู่ที่แตกต่างกันเป็นต้น

```

1  const express = require("express");
2  const router = express.Router();
3  const cors = require('cors');
4  var mysql = require("mysql");
5
6  router.use(cors())
7
8  // const User = require('../models/user')
9  process.env.SECRET_KEY = 'secret'
10
11 var con = mysql.createConnection({
12   host: "localhost",
13   user: "root",
14   password: "tnceao8617",
15   database: "mydb"
16 });
17 // DATABASE CONNECTION
18 con.connect(err => {
19   if (err) throw err;
20   console.log("Database For Build Connected!");
21 });
22
23 router.post("/database", (req, res, next) => {
24   con.query("CREATE DATABASE mydb", function(err, result) {
25     if (err) throw err;
26     console.log("Database created");
27   });
28 });
29
30 router.post("/table", (req, res, next) => {
31   var sql = 'CREATE TABLE user (userID int NOT NULL AUTO INCREMENT,Title VARCHAR(255),Firstname
32   con.query(sql, function(err, result) {
33     if (err) throw err;
34     console.log("Table 'user' created");
35     res.send("Table 'user' created");
36   });
37 });
38
39 router.post("/conname", (req, res, next) => {
40   var sql = 'CREATE TABLE conference (conferenceID int NOT NULL AUTO_INCREMENT,conferenceName
41   con.query(sql, function(err, result) {
42     if (err) throw err;
43     console.log("Table 'conference' created");
44     res.send("Table 'conference' created");
45   });
46 });
47
48 router.post("/memberConfer", (req, res, next) => {
49   var sql = 'CREATE TABLE memberConfer (memberID int NOT NULL AUTO_INCREMENT,email VARCHAR(255)
50   con.query(sql, function(err, result) {
51     if (err) throw err;
52     console.log("Table 'memberConfer' created");
53     res.send("Table 'memberConfer' created");
54   });
55 });
56
57 module.exports = router;

```

รูปที่ 14 ข. ตัวอย่างโค้ด Backend (api/createTable)

จากข้างต้นจะเป็น โค้ดของ Backend ในไฟล์ createTable ซึ่งมีไว้สำหรับสร้างฐานข้อมูลเพื่อรองรับ ข้อมูลต่างๆ ที่ผู้ใช้งานจะใส่เข้ามา และใช้สำหรับการดึงข้อมูลไป โดยจะมีส่วนหลักๆ ดังนี้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 1 ส่วนของการ Import Package หรือ ส่วนต่างๆ ที่เราต้องการใช้งาน

```
1  const express = require("express");
2  const router = express.Router();
3  const cors = require('cors');
4  var mysql = require("mysql");
```

รูปที่ 14 ข. ตัวอย่างโค้ด Backend ส่วนที่ 1

ส่วนที่ 2 ส่วนของการ เชื่อมต่อ Database

```
11 var con = mysql.createConnection({
12   host: "localhost",
13   user: "root",
14   password: "tncea08617",
15   database: "mydb"
16 });
17 // DATABASE CONNECTION
18 con.connect(err => {
19   if (err) throw err;
20   console.log("Database For Build Connected!");
```

รูปที่ 15 ข. ตัวอย่างโค้ด Backend ส่วนที่ 2

ส่วนที่ 3 ส่วนของการ เขียน API เพื่อรองรับการเรียกใช้ โดยจะเห็นได้ว่าเราจะใช้คำสั่ง Router.post หรือ Router.get ตามด้วย path ที่ต้องการให้เรียก ยกตัวอย่างเช่น (“/ABCD...”) เมื่อทำการเรียกใช้ Backend ก็จะทำให้การโค้ดที่เราเขียนไว้ พร้อม response ค่าที่เราต้องการ return กลับไป

```
router.post("/database", (req, res, next) => {
  con.query("CREATE DATABASE mydb", function(err, result) {
    if (err) throw err;
    console.log("Database created");
  });
});
```

รูปที่ 16 ข. ตัวอย่างโค้ด Backend ส่วนที่ 3

ส่วนที่ 4 ส่วนของการ export เพื่อนำไปเรียกใช้ในไฟล์อื่น ๆ ในที่นี้เมื่อทำการ exports แล้ว จะถูกนำไปเรียกใช้งานที่ app.js เพื่อรอการเรียกใช้งานเมื่อมี path ตรงกับที่เราสร้างไว้

```
57 module.exports = router;
```

รูปที่ 17 ข. ตัวอย่างโค้ด Backend ส่วนที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **node_modules** เป็น โฟลเดอร์ที่เก็บข้อมูล Package และ เครื่องมือสำคัญต่าง ที่ทำให้ตัว Project Backend นี้ สามารถรันใช้งานได้ โดย ข้อมูลในโฟลเดอร์นี้ห้ามถูกแก้ไขเด็ดขาดหรือลบออก อย่างผิดวิธีโดยเด็ดขาด

- **public** เป็น Folder ที่ใช้สำหรับเก็บรูปภาพหรือไฟล์ที่ทำการอัปโหลดมาจาก Frontend
- **.gitignore** เป็นไฟล์ที่เอาไว้ระบุ โฟลเดอร์ที่ไม่ต้องการ Upload ลงไปใน git ด้วย
- **App.js** เป็นไฟล์ศูนย์กลางที่จะรวบรวมรวมข้อมูลการ Route และ API ที่เขียนขึ้นมาไว้

```

1  const express = require('express');
2  const app = express();
3  const morgan = require('morgan');
4  const bodyParser = require('body-parser');
5  const cors = require('cors');
6
7  const User = require('./api/user');
8  const createTable = require('./api/createTable');
9  const memberCon = require('./api/memberConference');
10 const paymentPaypal = require('./api/paypal');
11
12 app.use(morgan('dev'));
13 app.use(cors());
14 app.use(bodyParser.urlencoded({extended: false}));
15 app.use(bodyParser.json());
16
17 // Security for browser
18 app.use((req,res,next)=> {
19   res.header('Access-Control-Allow-Origin','*');
20   res.header('Access-Control-Allow-Headers',"Origin, X-Requested-With,Content-Type,Accept,Authorization")
21   if (req.method === 'OPTIONS'){
22     res.header('Access-Control-Allow-Methods','PUT,POST,PATCH,DELETE,GET');
23     return res.status(200).json({});
24   }
25   next();
26 });
27
28 app.use('/', User);
29 app.use('/sql',createTable)
30 app.use('/memberconference',memberCon)
31 app.use('/paypal',paymentPaypal)
32
33 app.use((req,res,next)=> {
34   const error = new Error('Not Found');
35   error.status(404);
36   next(error);
37 })

```

รูปที่ 18 ข. ตัวอย่างโค้ด app.js (Backend)

- **Package.json** เป็น ไฟล์ที่คอยเก็บรายชื่อของเครื่องมือที่เราโหลดเข้ามาใช้งานใน Project นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Server.js** เป็นไฟล์ที่ใช้สำหรับรัน server backend เพิ่มที่ให้ผู้สามารถรองรับการใช้งานจาก
ที่อื่น

```

1  const http = require('http');
2  const app = require('./app');
3
4  const port = process.env.PORT || 5000;
5  const server = http.createServer(app);
6
7
8  console.log('Starting server at localhost:'+port);
9  server.listen(port);

```

รูปที่ 19 ข. ตัวอย่างโค้ด server.js

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค.
คู่มือการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานระบบนั้น สามารถแบ่งประเภทของการใช้งานได้เป็น 2 ประเภท ได้แก่ ผู้เข้าร่วมงาน (Attendee) และ ผู้จัดงาน (Organizer) ซึ่งผู้ใช้งานแต่ละประเภทจะมีฟังก์ชันและขั้นตอนที่แตกต่างกัน ซึ่งมีรายละเอียดดังต่อไปนี้

1. การใช้งานแบบผู้เข้าร่วมงาน (Attendee)

สำหรับผู้ใช้งานประเภทนี้ จะต้องทำการลงทะเบียนข้อมูลส่วนตัวก่อนเป็นอันดับแรก โดยมีขั้นตอนดังนี้

รูป 1ค. หน้าจอการ log-in เข้าสู่ระบบ

เมื่อเข้าสู่หน้าจอลงทะเบียนใช้งานระบบ ให้กด Create Account เพื่อทำการลงทะเบียนข้อมูลส่วนตัวและ Email สำหรับการเข้าสู่ระบบ โดยจะมีให้กรอกข้อมูลต่าง ๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Participant detail:

* Title

* Last name / Family name

* First name

Middle name

Affiliation

Address line 1

Address line 2

City

State

* Country

Postcode / ZIP code

* Phone

* Fax

* E-mail address

* Password

* Confirm Password

รูป 2ค. หน้าจอกรกรองข้อมูลส่วนตัว

Invoice detail:

Name of the recipient of the invoice

Recipient's VAT ID

Please provide the invoice recipient's address

Address line 1

Address line 2

City

State

Country

Postcode / ZIP code

รูป 3ค. หน้าจอกรกรองข้อมูลสำหรับการออกใบเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้เข้าร่วมงานจะต้องกรอกข้อมูลตามที่กำหนดเอาไว้ ซึ่งจะต้องกรอกข้อมูลในช่องที่มีเครื่องหมาย * เอาไว้ เมื่อผู้สมัครทำการยืนยันการสมัคร ระบบจะทำการพาผู้สมัครกลับเข้าหน้า log-in โดยผู้สมัครจะต้องใช้ Email และ Password ที่สมัครเอาไว้ข้างต้นในการเข้าสู่ระบบต่อไป

ในกรณีที่ผู้สมัครลืมรหัสผ่าน ผู้สมัครสามารถกดที่ Forgot Password? เพื่อเข้าหน้าเปลี่ยนรหัสผ่านได้

รูป 4ค. หน้าจอการเปลี่ยนรหัสผ่าน

ผู้สมัครจะต้องกรอก Email และ เบอร์ โทรศัพท์ ที่ถูกต้องเหมือนกับตอนที่สมัครในครั้งแรก จากนั้นจะสามารถทำการเปลี่ยนรหัสผ่านได้ โดยระบบจะพากลับไปหน้า log-in หลังจากกด Finish เพื่อยืนยันกลับเปลี่ยนรหัสผ่านเสร็จสิ้น

เมื่อผู้สมัครทำการ log-in เข้าสู่ระบบ Attendee ระบบจะทำการแสดงฟังก์ชันทั้งหมดที่ผู้เข้าร่วมงานสามารถทำได้ โดยมีทั้งหมดดังนี้

1.1) ลงทะเบียนงานประชุม (Active Conference)

Active Conference(s)

#	Conferences	Start Date	Finish Date	Payment Deadline
1	ACIIDS2021	13-February-2021	13-March-2021	13-December-2020
2	Ninja World War	13-March-2020	15-March-2020	11-January-2020

[MORE](#)

Conference: ACIIDS2021

* Participation type

* Registration type

Conference Date: 13 February 2021 - 13 March 2021

Payment Information :

Early Deadline:13 September 2020 Final Deadline:13 December 2020

Registration :

Early Registration: 120 Member Early Registration:100

Late Registration: 140 Member Late Registration:110 Student Late Registration:50

Other Information Price(EUR) :

Visitor Price:30 Extra Dinner Ticket:12 Additional Ticket:25

Additional registration detail:

Extra Ticket for your family

Extra Dinner Ticket

รูป 5ก. หน้าจอลงทะเบียนงานประชุม (Active Conference)

ผู้เข้าร่วมงานประชุมสามารถเลือกงานประชุมที่ขึ้นมาได้ดังรูป โดยเมื่อเลือกงานประชุมแล้วระบบจะทำการแสดงข้อมูลต่าง ๆ ของงาน เช่น วันจัดงาน , เวลาในการลงทะเบียน หรือราคาในการลงทะเบียนต่างๆ เป็นต้น ซึ่งผู้เข้าร่วมจะต้องเลือกประเภทการลงทะเบียนด้วยตัวเองในช่อง Participation type หรือประเภทผู้สมัคร ว่าเป็นผู้เขียนหรือผู้ช่วยเขียน และในช่อง Registration type หรือ ประเภทเวลาในการสมัคร ว่าเป็นแบบ ลงทะเบียนล่วงหน้า , ทั่วไป หรือล่าช้า เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2) เปลี่ยนข้อมูล (Changing Information)

Changing Information

Changing personal and billed information.

Be Careful!!
All of information that you change, it will effect to receipt and confereee data.

Personal Information

Refund Conference

Change Receipt

Change Contact Info

Tax Information

Memberships

For starting, select a choise.

รูป 6ค. หน้าจอเปลี่ยนข้อมูล

Personal Information

Refund Conference

Change Receipt

Change Contact Info

Tax Information

Memberships

Participant detail:

Title	Mr.
Last name / Family name	boonchoen
First name	thawin
Middle name	naruto
Affiliation	harvard
Address line 1	2132
Address line 2	asd
City	bangkok
State	bangkok
Country	Thailand
Postcode / ZIP code	10310
Phone	0945497733
Fax	0945497733

Cancel
Confirm

รูป 7ค. หน้าจอเปลี่ยนข้อมูลส่วนตัว

ผู้สมัครสามารถเปลี่ยนข้อมูลต่างๆ ของตัวเองได้ เช่น ข้อมูลส่วนตัวหรือข้อมูลใบเสร็จ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3) พิมพ์ใบเสร็จ (invoice)

Invoice

KMIT Ladkrabang Co., Ltd. (on behalf of "The 12th Asian Conference on Intelligent Information and Database Systems (ACIIDS 2020)")
 1 Soi Chalongkrung 1, Ladkrabang,
 Bangkok, Thailand , 10520
 VAT ID: 0105555155467
Recipient:Thawwin
Recipient VAT ID:11101010
Invoice #ACIIDS_KMITL_ID_2
Date:2020-03-03T14:31:12.000Z

No.	Description	Unit Cost (EUR)	Quantity	Price (EUR)
1	Registration fee for "ACIIDS2021" Registration ID: R0062 Participant type: "Author" Participant's name: thawwin	600	1	600
2	Registration type: "Member Late registration"	110	1	110
Extra: Total				710
Amount Paid				0.00
Balance Due				710

Payment Terms

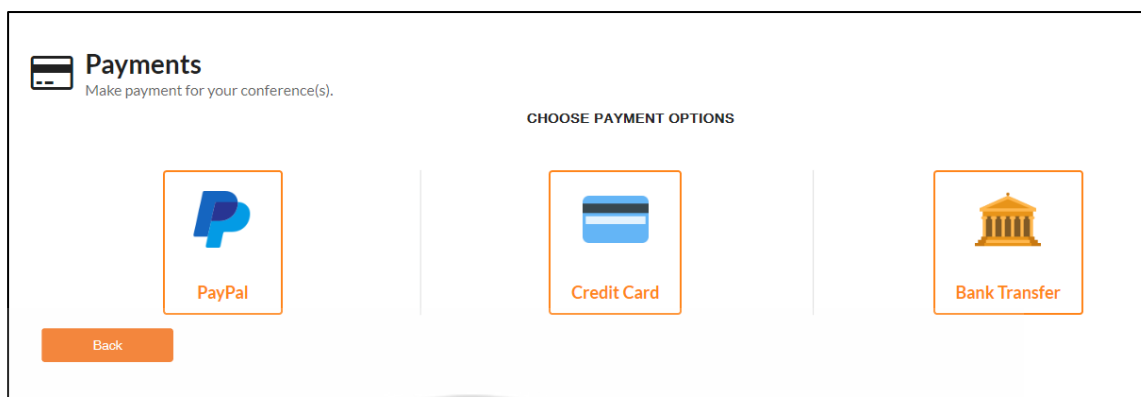
- Payment can be made by credit cards (online) or bank transfer.
- Account detail for bank transfer:
 Bank name: SIAM COMMERCIAL BANK PCL
 SWIFT Code: SICOTHBK
 Account no.: 414-050200-5 / Account name: KMIT LADKRABANG COMPANY LIMITED
 Branch code: 088 / Address: 3 MU2 CHALONGKRUNG RD., LADKRABANG
 City: BANGKOK / Country: THAILAND / Tel. [+66 2544 1000](tel:+6625441000)
 Reference/Message: ACIIDS R0062

รูป 8ค. รูปแบบใบเสร็จงานวิชาการ

ผู้สมัครสามารถปรับใบเสร็จการลงทะเบียนงานวิชาการที่ลงทะเบียนได้ โดยใบเสร็จจะประกอบด้วย ราคาของประเภทการลงทะเบียน และเวลาการลงทะเบียน ซึ่งระบบจะคำนวณราคาสุทธิออกมาให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4) ระบบชำระเงิน(Payment)



รูป 9ค. หน้าจอรระบบชำระเงิน(Payment)

ผู้สมัครสามารถชำระเงินสำหรับการประชุมวิชาการได้ทั้งหมด 3 ช่องทาง คือ PayPal , Credit Card และ Bank Transfer โดย 2 แบบแรก หากผู้สมัครเลือก ระบบจะทำการredirect เข้าไปที่ระบบของPayPalหรือCredit Card ข้างต้น และทำการชำระเงินภายในระบบนั้นๆ แต่ถ้าผู้สมัครเลือกโอนเงินผ่านBank จะต้องทำการยืนยันข้อมูลต่างๆดังนี้

 A screenshot of a web interface titled "Payments" with the subtitle "Make payment for your conference(s)". The form contains several fields: "Conference:" with a dropdown menu showing "Conference ID"; "Slip Photo:" with a text input field and a "Browse..." button; "Payment Date:" with a text input field showing "ex. 11/01/2020"; "Payment Time:" with a text input field showing "ex. 23:59"; and "Amount:" with a text input field showing "(EUR)". At the bottom of the form, there are two buttons: "BACK" and "UPLOAD".

รูป 10ค. หน้าจอรระบบชำระเงินด้วยการโอนเงินผ่านธนาคาร

ผู้สมัครจะต้องกรอกข้อมูลตาม Slip ที่ได้จากธนาคารตามความเป็นจริง เพื่อให้ผู้จัดงานสามารถเช็ค และเปลี่ยนสถานะให้ผู้สมัครได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5) ประวัติการลงทะเบียน (History)

ID	Conferences	Start Date	Finish Date	Payment Deadline	Payment Method	Payment Status
1	ACIIDS2021	13-February-2021	13-March-2021	13-December-2020	None	None
2	Ninja World War	13-March-2020	15-March-2020	11-January-2020	Bank Transfer	in Process

รูป 11ค. หน้าจอประวัติการลงทะเบียน (History)

ผู้สมัครสามารถติดตามการลงทะเบียนต่างๆที่เคยลงไปได้ เช่น วันที่งานจะเริ่ม, Deadline ในการชำระเงิน หรือสถานการณ์ชำระเงิน เป็นต้น โดยสำหรับสถานการณ์ชำระเงินนั้น ทางระบบจะแสดงว่าผู้สมัครทำการชำระเงินในรูปแบบไหน เช่น Bank Transfer หรือ PayPal เป็นต้น และถ้าหากผู้จัดงานยังไม่ได้ยืนยันว่า มีการชำระเงิน จะแสดงเป็น In Process แต่ถ้ามีการยืนยันแล้ว จะเปลี่ยนเป็น Complete

2. การใช้งานแบบผู้จัดงาน (Organizer)

การเข้าใช้งานระบบแบบผู้จัดงานจะแตกต่างกับแบบผู้เข้าร่วมงาน โดยที่ ID และ Password จะถูกกำหนดมาให้โดยผู้สร้างระบบ และผู้จัดงานไม่สามารถแก้ไขเกี่ยวกับ ID หรือ Password ได้

เมื่อผู้จัดงานทำการ log-in เข้าสู่ระบบ Organizer ระบบจะทำการแสดงฟังก์ชันทั้งหมดที่ผู้เข้าร่วมงานสามารถทำได้ โดยมีทั้งหมดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1) ภาพรวมของงานประชุม (Overview)

Overview

ACIDS2021

Conference Status

- Event Start Date : 13 February 2021
- Early Registration price (EUR): 120
- Payment Acceptance: Yes

Conference Data

- All of Attendee : 250
- All of Paper : 89
- Completed Documents (Papers) : 50
- Completed Payment (Papers) : 82

Conference Options:

Create Conference

รูป 12ค. หน้าจอภาพรวมของงานประชุม (Overview)

เมื่อผู้จัดงานทำการlog-inเสร็จแล้ว ระบบจะทำการแสดง overview ของงานประชุมต่างๆในหน้าแรกระบบ เช่น วันจัดงาน , จำนวนผู้เข้าร่วม หรือ จำนวนผู้ที่ทำการชำระเงินเรียบร้อยแล้ว เป็นต้น และผู้จัดงานสามารถเข้าถึงฟังก์ชันสร้างงานประชุมได้ โดยการกด create conference ด้านล่าง

2.2) ระบบสร้างงานประชุม(Create Conference)

Create a Conference

Conference Name:

Starting Date:

Finishing Date:

Early Birth Deadline:

Payment Deadline:

Registration Rate:

Early Birth Registration	Member Early Birth Registration	Regular Late Registration
EUR Amount <input type="text" value=".00"/>	EUR Amount <input type="text" value=".00"/>	EUR Amount <input type="text" value=".00"/>
Member Late Registration	Student Late Registration	Visitor (Non-Paper)
EUR Amount <input type="text" value=".00"/>	EUR Amount <input type="text" value=".00"/>	EUR Amount <input type="text" value=".00"/>
Extra Registration:	1-Day Pass City Tour	
EUR Amount <input type="text" value=".00"/>	EUR Amount <input type="text" value=".00"/>	

Clear & Reload

รูป 13ค. หน้าจกระบบสร้างงานประชุม(Create Conference)

ผู้จัดงานสามารถสร้างงานประชุมใหม่ขึ้นมาได้ โดยจะต้องกรอกข้อมูลทั้งหมดลงในช่องว่าง เพราะระบบจะนำข้อมูลทั้งหมดนี้ไปใช้ในการคำนวณราคาและข้อมูลต่างๆต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3) ระบบแก้ไขงานประชุม

Conferences

Beware to config conference data.
It will effect for payment rate and attendee accounts.

ID	Conferences	Start Date	Finish Date	Payment Deadline	Active Check	Delete Conference
1	ACIIDS2021	13-February-2021	13-March-2021	13-December-2020	Enable	Delete
2	Ninja World War	13-March-2020	15-March-2020	11-January-2020	Enable	Delete
7	Iron Fist Tournament	16-March-2022	16-May-2023	16-February-2022	Disable	Delete

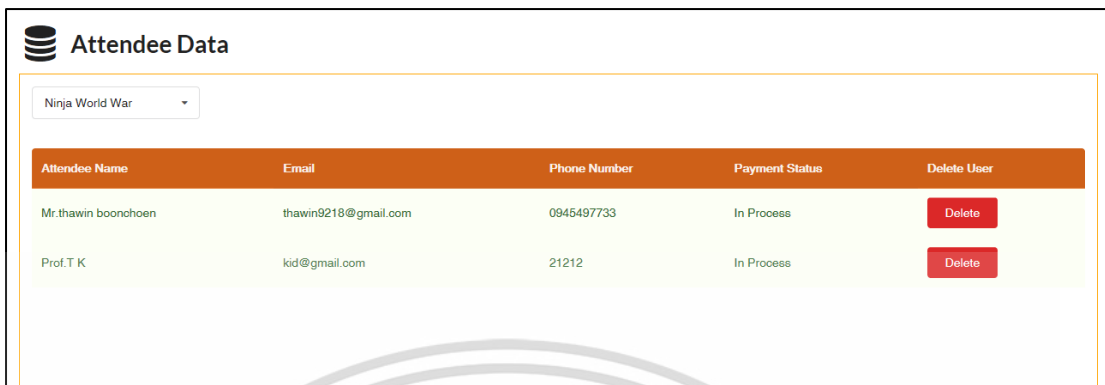
Conference Options:

Create Conference

รูป 14ค. หน้าจอระบบแก้ไขงานประชุม

ในฟังก์ชันผู้จัดงานจะสามารถเข้ามาแก้ไขข้อมูลต่างๆของงานประชุมทั้งหมดได้โดยกดตรงชื่องานประชุมนั้นๆ แล้วระบบจะredirectเข้าไปที่หน้าข้อมูลของงานประชุมที่เราเลือก และผู้จัดงานสามารถเลือกที่จะให้งานประชุมให้ยัง active หรือ ปิดรับสมัครอยู่ได้ในช่อง Active Check ถ้าหากยังเปิดรับอยู่ งานประชุมนั้นๆ จะขึ้นว่า Enable แต่ถ้าปิดรับสมัคร จะขึ้นว่า Disable ซึ่งสามารถกดเปลี่ยนสถานะเหล่านี้ได้ตลอด ผู้จัดงานยังสามารถลดงานประชุมที่ไม่ต้องการหรือจบลงไปแล้วได้ โดยการกด Delete ในช่องสุดท้าย

2.4) ระบบตรวจสอบข้อมูลผู้สมัคร(Attendee Data)



Attendee Name	Email	Phone Number	Payment Status	Delete User
Mr.thawin boonchoen	thawin9218@gmail.com	0945497733	In Process	Delete
Prof.T K	kid@gmail.com	21212	In Process	Delete

รูป 15ค. หน้าจกระบบตรวจสอบข้อมูลผู้สมัคร(Attendee Data)

ระบบนี้จะอนุญาตให้ผู้จัดงานตรวจสอบข้อมูลรวมไปถึงการแก้ไขข้อมูลต่างๆของผู้เข้าร่วมงานรวมถึงลบผู้สมัครของแต่ละงานประชุมได้ โดยกดที่ชื่อของผู้สมัคร แล้วระบบจะredirect ไปที่หน้าข้อมูลของคนนั้นๆ ซึ่งระบบนี้จะสามารถเข้าถึงระบบยืนยันการชำระเงินของผู้เข้าร่วมงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

“ทำความรู้จักกับ Redux”

[ระบบออนไลน์]. แหล่งที่มา <https://microbenz.in.th/ทำความรู้จักกับ-redux-แบบฉบับย่อแล้วย่ออีก-b464808aca12>

“มาเขียน ES6 กัน”

[ระบบออนไลน์]. แหล่งที่มา <https://microbenz.in.th/ปี-2017-แล้ว-มาเขียน-es6-กัน-9dede81e30da>

“Microsoft SQL Server คืออะไร”

[ระบบออนไลน์]. แหล่งที่มา <https://www.9experttraining.com/articles/microsoft-sql-server-คืออะไร>

“Payment Gateway คืออะไร”

[ระบบออนไลน์]. แหล่งที่มา <https://rabbitfinance.com/credit-card/glossary/payment-gateway>