รายงานวิจัยฉบับสมบูรณ์

โครงการ การพัฒนาแพลตฟอร์มสำหรับสนับสนุนบริการงานประชุมวิชาการ

Development of Conference Management Platform

นายวัชรพงศ์ อยู่ขวัญ

นางอธิตา อ่อนเอื้อน

โครงการประเภทเงินรายได้

คณะวิทยาการสารสนเทศ

ประจำปีงบประมาณ พ.ศ. 2566

มหาวิทยาลัยบูรพา

สัญญาเลขที่ ๐๐๗ฝ๒๕๖๖

รายงานวิจัยฉบับสมบูรณ์

โครงการ การพัฒนาแพลตฟอร์มสำหรับสนับสนุนบริการงานประชุมวิชาการ

Development of Conference Management Platform

นายวัชรพงศ์ อยู่ขวัญ

นางอธิตา อ่อนเอื้อน

กรกฎาคม พ.ศ. ๒๕๖๗

สารบัญรูปภาพ

หน้าที่

ภาพที่ 1 Fast API 20
ภาพที่ 2 DrizzleORM 21
ภาพที่ 3 ขั้นตอนการทำงานของระบบชำระเงิน 25
ภาพที่ 4 หน้าเว็บลงชื่อเข้าใช้สำหรับผู้ดูแลระบบ 26
ภาพที่ 5 หน้าเว็บแสดงรายการงานประชุมวิชาการ (Conference) หรืองานกิจกรรมที่กำลังใช้งานอยู่ 27
ภาพที่ 6 หน้าเว็บสำหรับแสดงรายละเอียดงานประชุมวิชาการ 28
ภาพที่ 7 หน้าเว็บสำหรับแก้ไขรายละเอียดงานประชุมวิชาการ 29
ภาพที่ 8 หน้าเว็บสำหรับเพิ่ม E-mail สำหรับงานประชุมวิชาการ 30
ภาพที่ 9 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการ 31
ภาพที่ 10 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการรูปแบบ Spreadsheet .. 32
ภาพที่ 11 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการแบบครั้งละบทความ 33
ภาพที่ 12 หน้าเว็บสำหรับเพิ่มรายชื่อประเภทของบทความพร้อมทั้งกำหนดราคาของบทความแต่ละประเภทในงานประชุมวิชาการแบบครั้งละบทความ 33
ภาพที่ 13 หน้าเว็บสำหรับเพิ่มประเภทการชำระเงินในงานประชุมวิชาการแบบครั้งละบทความ 34
ภาพที่ 14 หน้าเว็บสำหรับกำหนดค่า Parameter สำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต 34
ภาพที่ 15 หน้าเว็บสำหรับเสดง URL สำหรับการลงทะเบียน 35
ภาพที่ 16 หน้าเว็บสำหรับแสดง Transactions การชำระเงิน 35
ภาพที่ 17 หน้าเว็บสำหรับลงทะเบียนสำหรับเจ้าของบทความ 36
ภาพที่ 18 หน้าเว็บสำหรับออกใบเสร็จรับเงิน 36
ภาพที่ 19 หน้าเว็บสำหรับระบุรายละเอียดการออกใบเสร็จรับเงิน 37
ภาพที่ 20 หน้าเว็บสำหรับตรวจสอบและยืนยันข้อมูล 37
ภาพที่ 21 หน้าเว็บสำหรับแสดงตัวอย่างใบเสร็จรับเงิน 38
ภาพที่ 22 หน้าเว็บ API KTB Fast Pay 39
ภาพที่ 23 ที่ Gateway ของผู้ให้บริการบัตรของผู้ลงทะเบียน 39
ภาพที่ 24 หน้าเว็บสำหรับขอรหัส OTP ของผู้ให้บริการ 40
ภาพที่ 25 หน้าเว็บสำหรับยืนยันผลการชำระเงิน 40
ภาพที่ 26 ตัวอย่าง Email ใบเสร็จรับเงินชั่วคราว 41
ภาพที่ 27 หน้าเว็บแสดงรายละเอียดของ Transactions สำหรับเจ้าหน้าที่ฝ่ายการเงิน 41

กิตติกรรมประกาศ

ข้าพเจ้าขอขอบคุณคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ที่มอบทุนสนับสนุนในครั้งนี้ ทุนที่ได้รับทำให้โครงการวิจัยของข้าพเจ้าสำเร็จลุล่วงไปได้ด้วยดี ข้าพเจ้าขอแสดงความขอบคุณในความช่วยเหลือและการสนับสนุนที่มีค่าจากทางคณะ ซึ่งมีส่วนสำคัญในการพัฒนาผลงานวิจัยนี้

ความเอื้อเฟื้อและการสนับสนุนจากคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ทำให้ข้าพเจ้าได้รับโอกาสในการศึกษาและวิจัยในด้านที่สนใจอย่างเต็มที่ ข้าพเจ้าหวังเป็นอย่างยิ่งว่าผลงานวิจัยนี้จะเป็นประโยชน์และสามารถนำไปต่อยอดเพื่อสร้างสรรค์ผลงานใหม่ๆ ในอนาคต

ด้วยความเคารพอย่างสูง

วัชรพงศ์ อยู่ขวัญ

บทคัดย่อ

โครงการวิจัยพัฒนาระบบชำระเงินค่าลงทะเบียนงานประชุมวิชาการเป็นความพยายามในการปรับปรุงกระบวนการจัดการการชำระเงินให้มีประสิทธิภาพมากยิ่งขึ้น ผ่านการวางแผนและการพัฒนาที่คำนึงถึงประสบการณ์และความพึงพอใจของผู้ใช้งานเป็นสำคัญ โครงการนี้เริ่มต้นจากการวิเคราะห์ปัญหาและความต้องการของผู้เข้าร่วมงานประชุมวิชาการ จากนั้นดำเนินการพัฒนาและทดสอบระบบเพื่อให้มั่นใจว่าระบบที่พัฒนาขึ้นนั้นสามารถตอบสนองความต้องการได้อย่างครบถ้วน ในขั้นตอนการวิเคราะห์ ทีมงานได้สำรวจและรวบรวมข้อมูลจากผู้ใช้งานเพื่อทำความเข้าใจถึงปัญหาที่พวกเขาประสบในการชำระเงินค่าลงทะเบียน เช่น ความล่าช้า ความไม่สะดวกในการใช้งาน และความไม่มั่นใจในความปลอดภัยของข้อมูลส่วนตัว ข้อมูลเหล่านี้ถูกนำมาใช้ในการออกแบบระบบใหม่ที่มีความทันสมัยและตอบสนองต่อความต้องการของผู้ใช้งาน

ระบบชำระเงินค่าลงทะเบียนที่พัฒนาขึ้นใหม่มีคุณสมบัติที่สำคัญหลายประการ ได้แก่ การดำเนินการชำระเงินที่รวดเร็วและมีประสิทธิภาพ การรักษาความปลอดภัยของข้อมูลส่วนตัว การยืนยันการชำระเงินและการแจ้งเตือนที่ชัดเจน นอกจากนี้ ระบบยังถูกออกแบบให้ใช้งานง่าย ไม่ซับซ้อน เพื่อให้ผู้ใช้งานสามารถดำเนินการได้ด้วยตนเองอย่างสะดวกสบาย หลังจากการพัฒนาระบบเสร็จสิ้น ทีมงานได้ดำเนินการทดสอบระบบกับกลุ่มผู้ใช้งานจริงและเก็บรวบรวมข้อมูลความพึงพอใจ ผลการประเมินความพึงพอใจจากผู้ใช้งานพบว่าระบบที่พัฒนาขึ้นนั้นมีความพึงพอใจในระดับสูง แสดงให้เห็นว่าระบบสามารถตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างดี ทั้งในด้านความสะดวก ความรวดเร็ว และความปลอดภัย อย่างไรก็ตาม ยังมีบางด้านที่ต้องการการปรับปรุง เช่น การช่วยเหลือและการสนับสนุนจากทีมงาน รวมถึงความสามารถในการแก้ไขปัญหาที่พบ จากการประเมินพบว่าผู้ใช้งานยังต้องการการช่วยเหลือจากทีมงานเพิ่มเติมในกรณีที่เกิดปัญหาในการใช้งาน การปรับปรุงในด้านนี้จะช่วยเสริมสร้างความพึงพอใจของผู้ใช้งานได้ในอนาคต

Abstract

The research project for developing the registration payment system for academic conferences is an effort to improve the efficiency of payment processing. This is achieved through planning and development that prioritize the experience and satisfaction of users. The project begins with analyzing the problems and needs of conference attendees, followed by the development and testing of the system to ensure that it meets all user requirements.

In the analysis phase, the team conducted surveys and gathered data from users to understand the issues they faced during registration payment, such as delays, inconvenience, and concerns about the security of personal information. This data was used to design a modern system that addresses these user needs. The newly developed registration payment system has several key features, including fast and efficient payment processing, personal information security, clear payment confirmation, and notifications. Additionally, the system is designed to be user-friendly and straightforward, allowing users to easily complete transactions on their own.

After the system was developed, the team tested it with actual users and collected satisfaction data. The satisfaction survey results showed high levels of user satisfaction, indicating that the system effectively meets user needs in terms of convenience, speed, and security. However, there are still areas that require improvement, such as user support and problem resolution capabilities. The evaluation revealed that users still need additional support from the team in case of issues during usage. Improving these areas will enhance user satisfaction in the future. The collection and analysis of user satisfaction data will be highly beneficial for further refining and developing the system to fully meet user needs in the future, ensuring that the registration payment process for academic conferences is as smooth and efficient as possible. This development is a significant step in enhancing the efficiency and satisfaction of using the registration payment system for academic conferences and will help to promote a positive image of future academic conferences.

สารบัญ

หน้าที่

บทที่ 1 1
บทนำ 1
1.1 ที่มาและความสำคัญ 1
1.2 วัตถุประสงค์ 2
1.3 ประโยชน์ที่ได้รับ 3
บทที่ 2 4
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 4
2.1 ภาษาโปรแกรมคอมพิวเตอร์ 4
2.2 กรอบแนวคิดของโปแกรม (Programming Framework) 8
2.2.1 ประโยชน์ของการใช้เฟรมเวิร์ก 8
2.2.2 ประเภทของเฟรมเวิร์ก 9
2.2.3 กรอบแนวคิด Fast API 20
2.2.4 DrizzleORM 21
2.2.5 KTB FastPay 22
บทที่ 3 23
ขั้นตอนวิธีและการดำเนินงานพัฒนา 23
3.1 หน้าเว็บลงชื่อเข้าใช้สำหรับผู้ดูแลระบบ 25
3.2 หน้าเว็บแสดงรายการงานประชุมวิชาการ (Conference) หรืองานกิจกรรมที่กำลังใช้งานอยู่ 27
3.3 หน้าเว็บสำหรับแสดงรายละเอียดงานประชุมวิชาการ 28
3.4 หน้าเว็บสำหรับแก้ไขรายละเอียดงานประชุมวิชาการ 29
3.5 หน้าเว็บสำหรับเพิ่ม E-mail สำหรับงานประชุมวิชาการ 30
3.6 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการ 31
3.7 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการรูปแบบ Spreadsheet 32

3.8 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการแบบครั้งละบทความ..... 33
3.9 หน้าเว็บสำหรับเพิ่มรายชื่อประเภทของบทความพร้อมทั้งกำหนดราคาของบทความแต่ละประเภทในงานประชุมวิชาการแบบครั้งละบทความ 33
3.10 หน้าเว็บสำหรับเพิ่มประเภทการชำระเงินในงานประชุมวิชาการแบบครั้งละบทความ..... 34
3.11 หน้าเว็บสำหรับกำหนดค่า Parameter สำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต 34
3.12 หน้าเว็บสำหรับเสดง URL สำหรับการลงทะเบียน..... 35
3.13 หน้าเว็บสำหรับแสดง Transactions การชำระเงิน 35
3.14 หน้าเว็บสำหรับลงทะเบียนสำหรับเจ้าของบทความ..... 36
3.15 หน้าเว็บสำหรับระบุรายละเอียดการออกใบเสร็จรับเงิน..... 36
3.16 หน้าเว็บสำหรับแสดงตัวอย่างใบเสร็จรับเงิน 38
3.17 หน้าเว็บ API KTB Fast Pay..... 38
3.18 ตัวอย่าง Email ใบเสร็จรับเงินชั่วคราว 41
3.19 หน้าเว็บแสดงรายละเอียดของ Transactions สำหรับเจ้าหน้าที่ฝ่ายการเงิน 41
3.20 หน้าเว็บแสดงหน้าตรวจสอบสถานะของการลงทะเบียน..... 44
3.21 หน้าเว็บสำหรับแสดงสถานะของบทความ 45
บทที่ 4 48
ผลการดำเนินงาน 48
4.1 การทดสอบ User Acceptance Test (UAT) 48
4.2 ความพึงพอใจของผู้ใช้ 50
บทที่ 5 53
สรุปผลการดำเนินงาน 53
บรรณานุกรม 55
ประวัตินักวิจัย..... 56

สารบัญรูปภาพ

หน้าที่

ภาพที่ 1 Fast API 20
ภาพที่ 2 DrizzleORM 21
ภาพที่ 3 ขั้นตอนการทำงานของระบบชำระเงิน 25
ภาพที่ 4 หน้าเว็บลงชื่อเข้าใช้สำหรับผู้ดูแลระบบ 26
ภาพที่ 5 หน้าเว็บแสดงรายการงานประชุมวิชาการ (Conference) หรืองานกิจกรรมที่กำลังใช้งานอยู่ 27
ภาพที่ 6 หน้าเว็บสำหรับแสดงรายละเอียดงานประชุมวิชาการ 28
ภาพที่ 7 หน้าเว็บสำหรับแก้ไขรายละเอียดงานประชุมวิชาการ 29
ภาพที่ 8 หน้าเว็บสำหรับเพิ่ม E-mail สำหรับงานประชุมวิชาการ 30
ภาพที่ 9 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการ 31
ภาพที่ 10 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการรูปแบบ Spreadsheet .. 32
ภาพที่ 11 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการแบบครั้งละบทความ 33
ภาพที่ 12 หน้าเว็บสำหรับเพิ่มรายชื่อประเภทของบทความพร้อมทั้งกำหนดราคาของบทความแต่ละประเภทในงานประชุมวิชาการแบบครั้งละบทความ 33
ภาพที่ 13 หน้าเว็บสำหรับเพิ่มประเภทการชำระเงินในงานประชุมวิชาการแบบครั้งละบทความ 34
ภาพที่ 14 หน้าเว็บสำหรับกำหนดค่า Parameter สำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต 34
ภาพที่ 15 หน้าเว็บสำหรับเสดง URL สำหรับการลงทะเบียน 35
ภาพที่ 16 หน้าเว็บสำหรับแสดง Transactions การชำระเงิน 35
ภาพที่ 17 หน้าเว็บสำหรับลงทะเบียนสำหรับเจ้าของบทความ 36
ภาพที่ 18 หน้าเว็บสำหรับออกใบเสร็จรับเงิน 36
ภาพที่ 19 หน้าเว็บสำหรับระบุรายละเอียดการออกใบเสร็จรับเงิน 37
ภาพที่ 20 หน้าเว็บสำหรับตรวจสอบและยืนยันข้อมูล 37
ภาพที่ 21 หน้าเว็บสำหรับแสดงตัวอย่างใบเสร็จรับเงิน 38
ภาพที่ 22 หน้าเว็บ API KTB Fast Pay 39
ภาพที่ 23 ที่ Gateway ของผู้ให้บริการบัตรของผู้ลงทะเบียน 39
ภาพที่ 24 หน้าเว็บสำหรับขอรหัส OTP ของผู้ให้บริการ 40
ภาพที่ 25 หน้าเว็บสำหรับยืนยันผลการชำระเงิน 40
ภาพที่ 26 ตัวอย่าง Email ใบเสร็จรับเงินชั่วคราว 41
ภาพที่ 27 หน้าเว็บแสดงรายละเอียดของ Transactions สำหรับเจ้าหน้าที่ฝ่ายการเงิน 41

หน้าที่

ภาพที่ 28 หน้าเว็บแสดงการปฏิเสธการชำระเงิน..... 42
ภาพที่ 29 หน้าเว็บสำหรับยืนยันการออกใบเสร็จรับเงิน..... 43
ภาพที่ 30 หน้าเว็บแสดงหน้าตรวจสอบสถานะของการลงทะเบียน..... 44
ภาพที่ 31 หน้าเว็บสำหรับตรวจสอบสถานะการชำระเงิน..... 44
ภาพที่ 32 หน้าเว็บสำหรับแสดงสถานะของบทความ..... 45

สารบัญตาราง

หน้าที่

ตารางที่ 1 ผลการทดสอบ UAT 48
ตารางที่ 2 ผลการสำรวจความพึงพอใจ 50

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

คณะวิทยาการสารสนเทศมีการจัดกิจกรรมและโครงการต่าง ๆ มากมายที่เกี่ยวข้องกับการชำระเงิน เช่น การลงทะเบียนเรียน การเข้าร่วมสัมมนา การซื้อสื่อการเรียนรู้และอื่น ๆ ในปัจจุบันกระบวนการชำระเงินดังกล่าวส่วนใหญ่ยังคงเป็นแบบดั้งเดิม ที่ต้องใช้เวลาและแรงงานในการดำเนินการ ซึ่งอาจทำให้เกิดความล่าช้าและความผิดพลาดในการจัดการข้อมูล

กิจกรรมที่คณะวิทยาการสารสนเทศจัดขึ้นในแต่ละปีมีจำนวนมากและมีความหลากหลาย ตั้งแต่การจัดการเรียนการสอน การสัมมนาเชิงปฏิบัติการ งานวิจัย การจัดประชุมวิชาการ การจัดอบรม รวมถึงการจัดกิจกรรมเสริมทักษะทางวิชาการและวิชาชีพ ซึ่งล้วนแล้วแต่ต้องมีการจัดการการชำระเงินทั้งสิ้น นักศึกษาต้องทำการลงทะเบียนเรียนในแต่ละภาคการศึกษา และชำระค่าธรรมเนียมการศึกษา รวมทั้งการซื้อหนังสือและอุปกรณ์การเรียนรู้ต่าง ๆ นอกจากนี้ ยังมีการจัดการชำระเงินสำหรับการเข้าร่วมกิจกรรมพิเศษ เช่น การเข้าร่วมสัมมนาหรือเวิร์กช็อปที่คณะจัดขึ้น การชำระเงินสำหรับการสอบวัดผลหรือการทดสอบต่าง ๆ ตลอดจนการบริหารจัดการค่าธรรมเนียมอื่น ๆ ที่เกี่ยวข้องกับการศึกษา

กระบวนการชำระเงินแบบดั้งเดิมที่คณะใช้ในปัจจุบันยังคงมีข้อจำกัดหลายประการ ที่สำคัญคือการที่นักศึกษาต้องเดินทางมาชำระเงินด้วยตนเองที่สำนักงานหรือผ่านช่องทางการชำระเงินที่จำกัด ซึ่งขั้นตอนนี้ใช้เวลามากและอาจทำให้เกิดความยุ่งยากและความไม่สะดวกสำหรับนักศึกษา โดยเฉพาะในช่วงเวลาที่มีการลงทะเบียนเรียนหรือกิจกรรมอื่น ๆ ที่มีผู้เข้าร่วมจำนวนมาก นอกจากนี้ การจัดการข้อมูลการชำระเงินด้วยวิธีดั้งเดิมยังเพิ่มความเสี่ยงในการเกิดข้อผิดพลาด ไม่ว่าจะเป็นการบันทึกข้อมูลผิดพลาด การสูญหายของใบเสร็จรับเงิน หรือการตรวจสอบข้อมูลที่ไม่ครบถ้วนและไม่ถูกต้อง

ด้วยเหตุนี้ การพัฒนาระบบบริหารจัดการการชำระเงินผ่านระบบออนไลน์จะช่วยให้กระบวนการชำระเงินต่าง ๆ เป็นไปอย่างรวดเร็ว แม่นยำ และมีประสิทธิภาพมากขึ้น ระบบออนไลน์จะช่วยลดระยะเวลาและความซับซ้อนของกระบวนการชำระเงิน นักศึกษาสามารถดำเนินการชำระเงินได้ทุกที่ทุกเวลา ไม่จำเป็นต้องเดินทางมาชำระเงินด้วยตนเองที่สำนักงานคณะ ทำให้ประหยัดเวลาและลดค่าใช้จ่ายในการเดินทาง นอกจากนี้ ระบบออนไลน์ยังช่วยเพิ่มความถูกต้องในการบันทึกข้อมูล เนื่องจากข้อมูลจะถูกบันทึกและประมวลผลโดยอัตโนมัติ ทำให้ลดความเสี่ยงในการเกิดข้อผิดพลาด

การพัฒนาระบบบริหารจัดการการชำระเงินผ่านระบบออนไลน์ยังเป็นการส่งเสริมการใช้เทคโนโลยีสารสนเทศให้เกิดประโยชน์สูงสุดในกระบวนการทำงานของคณะ การนำเทคโนโลยีมาใช้ในการจัดการการชำระเงินไม่เพียงแต่ช่วยเพิ่มประสิทธิภาพในการทำงาน แต่ยังเป็นการสร้างประสบการณ์ที่ดีให้กับนักศึกษาและบุคลากร โดยทำให้กระบวนการต่าง ๆ เป็นไปอย่างราบรื่นและสะดวกสบาย นอกจากนี้ ระบบออนไลน์ยังสามารถสร้างรายงานและสถิติการชำระเงินที่เป็นประโยชน์ต่อการบริหารจัดการและการวางแผนในอนาคต ทำให้ผู้บริหารสามารถตัดสินใจได้อย่างมีข้อมูลรองรับ

การใช้ระบบบริหารจัดการการชำระเงินผ่านระบบออนไลน์ยังช่วยเสริมสร้างภาพลักษณ์ที่ดีให้กับคณะวิทยาการสารสนเทศในฐานะสถาบันที่ทันสมัยและมุ่งเน้นการใช้เทคโนโลยีในการพัฒนาการศึกษา ทั้งนี้ การพัฒนาระบบดังกล่าวต้องคำนึงถึงความปลอดภัยของข้อมูลและความเป็นส่วนตัวของผู้ใช้ โดยมีการใช้เทคโนโลยีที่ทันสมัยและมาตรฐานความปลอดภัยที่สูงสุด เพื่อให้ผู้ใช้มั่นใจในการทำธุรกรรมออนไลน์

สรุปได้ว่า การพัฒนาระบบบริหารจัดการการชำระเงินผ่านระบบออนไลน์สำหรับคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ไม่เพียงแต่จะช่วยเพิ่มประสิทธิภาพในการทำงาน ลดระยะเวลาและความซับซ้อนของกระบวนการชำระเงิน แต่ยังเป็นการส่งเสริมการใช้เทคโนโลยีสารสนเทศให้เกิดประโยชน์สูงสุดในกระบวนการทำงานของคณะ ทำให้การดำเนินงานเป็นไปอย่างราบรื่น สะดวกสบาย และมีความปลอดภัยสูงสุด ทั้งนี้เพื่อประโยชน์สูงสุดต่อบุคลากร นักศึกษา และสถาบันในภาพรวม

1.2 วัตถุประสงค์

1.2.1 เพิ่มประสิทธิภาพในการจัดการการชำระเงิน

- ลดระยะเวลาและขั้นตอนที่ซับซ้อนในการจัดการการชำระเงิน
- ลดความผิดพลาดที่อาจเกิดขึ้นจากการจัดการข้อมูลด้วยมือ

1.2.2 เพิ่มความสะดวกสบายให้แก่ผู้ใช้

- ให้บริการชำระเงินผ่านระบบออนไลน์ที่ใช้งานง่ายและสะดวกสบายสำหรับบุคลากรและนักศึกษารวมถึงผู้เข้าร่วมงานประชุมวิชาการหรืองานกิจกรรมต่าง ๆ
- สามารถดำเนินการชำระเงินได้ทุกที่ทุกเวลา ไม่จำเป็นต้องเดินทางมาชำระเงินด้วยตนเองที่สำนักงานคณะ
- ใช้เทคโนโลยีที่ทันสมัยและมาตรฐานความปลอดภัยสูงสุด เพื่อให้ผู้ใช้มั่นใจในการทำธุรกรรมออนไลน์

1.2.3 สนับสนุนการพัฒนาทักษะทางเทคโนโลยี

- ส่งเสริมให้มีโอกาสพัฒนาทักษะทางเทคโนโลยีสารสนเทศที่เกี่ยวข้องกับการใช้งานระบบบริหารจัดการออนไลน์

1.3 ประโยชน์ที่ได้รับ

1.3.1 ประสิทธิภาพในการจัดการที่เพิ่มขึ้น

- ลดระยะเวลาและความซับซ้อนของกระบวนการชำระเงิน ทำให้การดำเนินการต่าง ๆ รวดเร็วและมีประสิทธิภาพมากขึ้น
- ลดภาระงานของบุคลากรในการจัดการข้อมูลการชำระเงินแบบดั้งเดิม

1.3.2 ความสะดวกสบายของผู้ใช้:

- สามารถชำระเงินได้ทุกที่ทุกเวลา ผ่านระบบออนไลน์ ไม่ต้องเดินทางมาชำระเงินที่สำนักงาน
- ระบบออนไลน์ใช้งานง่ายและสามารถเข้าถึงได้ผ่านอุปกรณ์ต่าง ๆ เช่น คอมพิวเตอร์ โทรศัพท์มือถือ หรือแท็บเล็ต

1.3.3 ความปลอดภัย

- ใช้เทคโนโลยีและมาตรฐานความปลอดภัยสูงสุดในการปกป้องข้อมูลส่วนบุคคลและข้อมูลการทำธุรกรรม
- สร้างความมั่นใจให้กับผู้ใช้ในการทำธุรกรรมออนไลน์

1.3.4 การสนับสนุนการวางแผนและการตัดสินใจ:

- ระบบออนไลน์สามารถสร้างรายงานและสถิติที่เป็นประโยชน์ต่อการบริหารจัดการและการวางแผนในอนาคต
- ผู้บริหารสามารถใช้ข้อมูลที่ถูกต้องและทันสมัยในการตัดสินใจและวางกลยุทธ์ในการพัฒนาคณะ

1.3.5 เสริมสร้างภาพลักษณ์ที่ดีของคณะ

- ทำให้คณะวิทยาการสารสนเทศมีภาพลักษณ์ที่ทันสมัยและเป็นผู้นำในการใช้เทคโนโลยีในการจัดการและพัฒนาการศึกษา
- ส่งเสริมให้คณะเป็นที่ยอมรับในด้านการบริหารจัดการและการใช้เทคโนโลยีสารสนเทศในวงการการศึกษา

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

การพัฒนาระบบบริหารจัดการการชำระเงินออนไลน์สำหรับงานประชุมวิชาการและกิจกรรมต่าง ๆ อาศัยทฤษฎีระบบสารสนเทศ ทฤษฎีการจัดการเปลี่ยนแปลง ทฤษฎีการยอมรับเทคโนโลยี และทฤษฎีความปลอดภัยของข้อมูล เพื่อเพิ่มประสิทธิภาพ ลดความซับซ้อน และเพิ่มความสะดวกสบายในการทำธุรกรรมออนไลน์ งานวิจัยที่เกี่ยวข้องชี้ให้เห็นถึงประโยชน์ของระบบการชำระเงินออนไลน์ในการจัดการค่าใช้จ่าย การยอมรับเทคโนโลยีใหม่และมาตรการความปลอดภัยที่จำเป็น การนำระบบดังกล่าวมาใช้จะช่วยให้การจัดการงานประชุมวิชาการและกิจกรรมต่าง ๆ เป็นไปอย่างราบรื่น ปลอดภัย และทันสมัย ส่งเสริมทักษะทางเทคโนโลยีของผู้ใช้ และเสริมสร้างภาพลักษณ์ที่ดีให้กับสถาบันการศึกษา

2.1 ภาษาโปรแกรมคอมพิวเตอร์

การพัฒนาระบบบริหารจัดการการชำระเงินออนไลน์สำหรับงานประชุมวิชาการและกิจกรรมต่าง ๆ สามารถใช้ภาษาโปรแกรมคอมพิวเตอร์หลายภาษาเพื่อสร้างระบบที่มีประสิทธิภาพและปลอดภัยได้ โดยภาษาที่ถูกนำมาใช้เพื่อพัฒนาระบบ ฯ ได้แก่ HTML, CSS, และ JavaScript ใช้สำหรับการพัฒนาอินเทอร์เฟซผู้ใช้ (User Interface) และการแสดงผลบนเว็บ JavaScript frameworks ประกอบด้วย Vue.js และ Nuxt.js สามารถเพิ่มประสิทธิภาพและประสบการณ์การใช้งานของผู้ใช้ รวมถึง PHP ซึ่งเป็นภาษาเซิร์ฟเวอร์ที่ใช้สร้างและจัดการการทำงานของระบบหลังบ้าน (backend) ใช้ร่วมกับฐานข้อมูลเช่น MySQL หรือ PostgreSQL เพื่อจัดเก็บและดึงข้อมูล นอกจากนี้ระบบ ฯ มีการใช้งานภาษา Python ใช้สำหรับพัฒนาระบบหลังบ้านและสามารถสร้าง API ที่ปลอดภัยและมีประสิทธิภาพ รวมถึงการใช้งาน API Integration การรวม API จากผู้ให้บริการการชำระเงินของธนาคารกรุงไทย (KTB) ซึ่งเป็นผู้ใหบริการด้านการเงินหลักของมหาวิทยาลัยบูรพา

2.1.1 การำงานร่วมกันของ HTML, CSS และ Javascript

การทำงานร่วมกันของ HTML, CSS และ JavaScript เป็นพื้นฐานสำคัญในการพัฒนาเว็บแอปพลิเคชันที่ทันสมัยและมีประสิทธิภาพ ทั้งสามองค์ประกอบนี้ทำงานร่วมกันเพื่อสร้างประสบการณ์การใช้งานที่ดีและตอบสนองความต้องการของผู้ใช้

การสร้างเว็บแอปพลิเคชันที่มีประสิทธิภาพและมีความน่าสนใจต้องอาศัยการทำงานร่วมกันของ HTML, CSS และ JavaScript ทั้งสามองค์ประกอบนี้ทำงานร่วมกันเพื่อสร้างหน้าเว็บที่มีโครงสร้างชัดเจน, มีการจัดรูปแบบ

ที่ดี และสามารถโต้ตอบกับผู้ใช้ได้ ในส่วนต่อไปนี้จะอธิบายการทำงานของแต่ละองค์ประกอบและการผสมผสานการทำงานอย่างละเอียด

HTML (HyperText Markup Language)

ภาษา HTML ใช้สำหรับสร้างโครงสร้างและเนื้อหาของหน้าเว็บ เช่น ข้อความ รูปภาพ และลิงก์ HTML เป็นภาษามาร์กอัปที่ใช้สร้างโครงสร้างและเนื้อหาของหน้าเว็บ โดยใช้แท็ก (tags) เพื่อกำหนดองค์ประกอบต่าง ๆ เช่น ข้อความ, รูปภาพ, ลิงก์, ตาราง และฟอร์ม HTML เป็นพื้นฐานที่จำเป็นในการสร้างหน้าเว็บเพราะมันกำหนดว่าองค์ประกอบต่าง ๆ จะถูกวางที่ใดในหน้าเว็บ โครงสร้างพื้นฐานของ HTML HTML ประกอบด้วยเอกสาร HTML5 ที่เริ่มต้นด้วย <!DOCTYPE html> และตามด้วยโครงสร้างหลักที่ประกอบด้วย <html>, <head>, และ <body> ดังตัวอย่าง

```
<!DOCTYPE html>

<html>
<head>
  <title>My Web Page</title>
</head>
<body>
  <h1>Welcome to My Web Page</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

- DOCTYPE declaration: กำหนดว่าเอกสารนี้เป็น HTML5
- <html>: แท็กหลักที่ครอบคลุมทุกอย่างในหน้าเว็บ
- <head>: ส่วนนี้ใช้สำหรับเมตาดาต้า เช่น ชื่อเรื่อง (title), การเชื่อมโยงไฟล์ CSS และ JavaScript, และการกำหนดตัวอักษร
- <body>: ส่วนนี้ใช้สำหรับเนื้อหาหลักของหน้าเว็บ เช่น ข้อความ, รูปภาพ, และลิงก์

CSS (Cascading Style Sheets)

ภาษา CSS ใช้สำหรับการจัดรูปแบบและการออกแบบหน้าเว็บ เช่น สี ฟอนต์ และการจัดวางองค์ประกอบ CSS ใช้สำหรับการจัดรูปแบบและการออกแบบหน้าเว็บ เช่น สี, ฟอนต์, การจัดวางองค์ประกอบ, และการตอบสนองต่อการแสดงผลบนอุปกรณ์ต่าง ๆ (responsive design) CSS ทำให้หน้าเว็บดูสวยงามและเข้ากับการออกแบบที่ต้องการ การเชื่อมโยง CSS กับ HTML โดย CSS สามารถเชื่อมโยงกับ HTML ได้หลายวิธี เช่น การใส่ในแท็ก <style> ภายใน <head>, การใส่ในตัวโค้ด HTML โดยตรง (inline styles), และการเชื่อมโยงไฟล์ CSS ภายนอก

```
<!DOCTYPE html>
<html>
<head>
  <title>My Web Page</title>
  <style>
    body {
      background-color: #f0f0f0;
    }
    h1 {
      color: blue;
    }
    p {
      font-size: 16px;
    }
  </style>
</head>
<body>
  <h1>Welcome to My Web Page</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

JavaScript (JS)

ภาษา JavaScript ใช้สำหรับการเพิ่มฟังก์ชันและการโต้ตอบให้กับหน้าเว็บ เช่น การตรวจสอบข้อมูลแบบฟอร์ม การสร้างป๊อปอัป และการเปลี่ยนแปลงเนื้อหาแบบไดนามิก JavaScript เป็นภาษาสคริปต์ที่ใช้เพิ่มฟังก์ชันและการโต้ตอบให้กับหน้าเว็บ เช่น การตรวจสอบข้อมูลในฟอร์ม การสร้างป๊อปอัป การเปลี่ยนแปลงเนื้อหาแบบไดนามิก การควบคุมสไตล์และองค์ประกอบ และการจัดการเหตุการณ์ (event handling) การใช้ JavaScript ทำให้หน้าเว็บมีความสมบูรณ์และตอบสนองต่อการกระทำของผู้ใช้ได้

```
<!DOCTYPE html>
<html>
<head>
  <title>My Web Page</title>
  <style>
    body {
      background-color: #f0f0f0;
    }
    h1 {
      color: blue;
    }
    p {
      font-size: 16px;
    }
  </style>
</head>
<body>
  <h1>Welcome to My Web Page</h1>
  <p id="demo">This is a paragraph.</p>
  <button type="button" onclick="changeText()">Click Me!</button>
```

```
  <script>
    function changeText() {
      document.getElementById("demo").innerHTML = "You clicked the button!";
    }
  </script>
</body>
</html>
```

2.2 กรอบแนวคิดของโปแกรม (Programming Framework)

เฟรมเวิร์กหรือกรอบแนวคิดของการเขียนโปรแกรม (Programming Framework) เป็นชุดของเครื่องมือ, ไลบรารี, และแนวทางปฏิบัติที่ถูกออกแบบมาเพื่อช่วยให้นักพัฒนาซอฟต์แวร์สามารถสร้างแอปพลิเคชันได้อย่างมีประสิทธิภาพและประสิทธิผลมากขึ้น เฟรมเวิร์กเป็นโครงสร้างพื้นฐานที่นักพัฒนาสามารถใช้เพื่อสร้างและพัฒนาแอปพลิเคชันได้โดยไม่ต้องเริ่มต้นจากศูนย์

2.2.1 ประโยชน์ของการใช้เฟรมเวิร์ก

1. **การเร่งการพัฒนา (Development Speed)**: เฟรมเวิร์กมีโค้ดพื้นฐานและไลบรารีที่สามารถใช้ซ้ำได้ ซึ่งช่วยลดเวลาในการพัฒนาแอปพลิเคชัน
2. **การปรับปรุงประสิทธิภาพ (Improved Efficiency)**: เฟรมเวิร์กมักจะมีการจัดการกับเรื่องที่ซับซ้อน เช่น การจัดการฐานข้อมูล การรักษาความปลอดภัย และการจัดการเซสชัน ทำให้นักพัฒนาสามารถโฟกัสไปที่การสร้างฟีเจอร์ใหม่ ๆ ได้
3. **การลดข้อผิดพลาด (Error Reduction)**: เฟรมเวิร์กมักจะมีการทดสอบและตรวจสอบคุณภาพอย่างดี ทำให้โค้ดที่ใช้นั้นมีคุณภาพและลดข้อผิดพลาดในการพัฒนา
4. **การบำรุงรักษาและการปรับปรุง (Maintenance and Upgrades)**: เฟรมเวิร์กมีชุมชนผู้ใช้งานที่ช่วยกันพัฒนาและปรับปรุง ทำให้สามารถอัปเดตและบำรุงรักษาได้ง่าย
5. **การทำงานร่วมกัน (Collaboration)**: เฟรมเวิร์กมีโครงสร้างที่ชัดเจน ทำให้ทีมพัฒนาสามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพมากขึ้น

2.2.2 ประเภทของเฟรมเวิร์ก

1. **Web Development Frameworks**: ใช้ในการพัฒนาแอปพลิเคชันเว็บ ตัวอย่างเช่น:
    - **Django** (Python): เฟรมเวิร์กที่มีเครื่องมือครบครันสำหรับการพัฒนาเว็บแอปพลิเคชันด้วยภาษา Python
    - **Ruby on Rails** (Ruby): เฟรมเวิร์กที่เน้นการพัฒนาอย่างรวดเร็วด้วยแนวคิด Convention over Configuration
    - **Spring** (Java): เฟรมเวิร์กที่ช่วยในการพัฒนาแอปพลิเคชันองค์กร (Enterprise Applications) ด้วยภาษา Java
    - **Laravel** (PHP): เฟรมเวิร์กที่มีเครื่องมือหลากหลายสำหรับการพัฒนาเว็บด้วยภาษา PHP
    - **Angular** (JavaScript): เฟรมเวิร์กสำหรับการพัฒนาเว็บแอปพลิเคชันที่มีความซับซ้อนด้วยภาษา JavaScript
2. **Mobile Development Frameworks**: ใช้ในการพัฒนาแอปพลิเคชันมือถือ ตัวอย่างเช่น:
    - **React Native**: เฟรมเวิร์กจาก Facebook ที่ใช้ในการพัฒนาแอปพลิเคชันมือถือที่สามารถทำงานได้ทั้งบน iOS และ Android โดยใช้ภาษา JavaScript
    - **Flutter**: เฟรมเวิร์กจาก Google ที่ใช้ในการพัฒนาแอปพลิเคชันมือถือด้วยภาษา Dart ซึ่งมีประสิทธิภาพสูงและสามารถทำงานข้ามแพลตฟอร์มได้
    - **Xamarin**: เฟรมเวิร์กจาก Microsoft ที่ใช้ในการพัฒนาแอปพลิเคชันมือถือด้วยภาษา C# และ .NET
3. **Game Development Frameworks**: ใช้ในการพัฒนาเกม ตัวอย่างเช่น:
    - **Unity**: เฟรมเวิร์กที่ได้รับความนิยมสูงสุดสำหรับการพัฒนาเกมที่สามารถทำงานได้บนหลายแพลตฟอร์ม
    - **Unreal Engine**: เฟรมเวิร์กจาก Epic Games ที่มีเครื่องมือครบครันสำหรับการพัฒนาเกมที่มีกราฟิกและประสิทธิภาพสูง
    - **Godot**: เฟรมเวิร์กที่เป็นโอเพนซอร์สและมีความยืดหยุ่นสูงสำหรับการพัฒนาเกม
4. **Data Science Frameworks**: ใช้ในการวิเคราะห์ข้อมูลและสร้างโมเดลการเรียนรู้ของเครื่อง (Machine Learning) ตัวอย่างเช่น:
    - **TensorFlow**: เฟรมเวิร์กจาก Google ที่ใช้ในการพัฒนาและฝึกฝนโมเดลการเรียนรู้ของเครื่อง
    - **PyTorch**: เฟรมเวิร์กจาก Facebook ที่มีความยืดหยุ่นสูงและใช้งานง่ายสำหรับการพัฒนาโมเดลการเรียนรู้ของเครื่อง

- **Scikit-learn**: ไลบรารีที่มีเครื่องมือสำหรับการวิเคราะห์ข้อมูลและสร้างโมเดลการเรียนรู้ของเครื่อง

5. **Cross-platform Frameworks**: ใช้ในการพัฒนาแอปพลิเคชันที่สามารถทำงานได้บนหลายแพลตฟอร์ม ตัวอย่างเช่น:
   - **Electron**: เฟรมเวิร์กที่ใช้ในการพัฒนาแอปพลิเคชันเดสก์ท็อปด้วยเทคโนโลยีเว็บ (HTML, CSS, JavaScript)
   - **Qt**: เฟรมเวิร์กที่ใช้ในการพัฒนาแอปพลิเคชันที่สามารถทำงานได้บนหลายแพลตฟอร์มด้วยภาษา C++
   - **.NET**: เฟรมเวิร์กจาก Microsoft ที่สามารถใช้ในการพัฒนาแอปพลิเคชันเดสก์ท็อป, เว็บ, และมือถือ

เฟรมเวิร์กการเขียนโปรแกรมเป็นเครื่องมือที่สำคัญสำหรับนักพัฒนาซอฟต์แวร์ เนื่องจากช่วยให้การพัฒนาแอปพลิเคชันเป็นไปอย่างมีประสิทธิภาพและประสิทธิผลมากขึ้น เฟรมเวิร์กมีหลากหลายประเภทขึ้นอยู่กับการใช้งาน ไม่ว่าจะเป็นการพัฒนาเว็บแอปพลิเคชัน, แอปพลิเคชันมือถือ, เกม, การวิเคราะห์ข้อมูล, หรือแอปพลิเคชันข้ามแพลตฟอร์ม การเลือกใช้เฟรมเวิร์กที่เหมาะสมกับงานสามารถช่วยลดเวลาในการพัฒนาและเพิ่มคุณภาพของซอฟต์แวร์ที่สร้างขึ้นได้ โดยในโครงการนี้จะใช้ nuxt.js และ bootstrap สำหรับพัฒนาส่วนติดต่อผู้ใช้และใช้งาน Fast API ของภาษา Python เพื่อการเชื่อมต่อฐานข้อมูล

2.3.1 NuxtJS Framewotk

Nuxt.js เป็นเฟรมเวิร์คที่พัฒนาอยู่บน Vue.js โดยออกแบบมาเพื่อสร้างแอปพลิเคชัน Vue.js ที่มีประสิทธิภาพและโครงสร้างที่ดีมากขึ้น Nuxt.js ให้ความสำคัญกับการพัฒนาแอปพลิเคชันฝั่งเซิร์ฟเวอร์ (Server-Side Rendering - SSR) รวมถึงการสร้างแอปพลิเคชันหน้าเดียว (Single Page Applications - SPA) และการพัฒนาแอปพลิเคชันแบบไร้เซิร์ฟเวอร์ (Static Site Generation - SSG)

คุณสมบัติเด่นของ Nuxt.js

1. **Server-Side Rendering (SSR)**: Nuxt.js ช่วยให้แอปพลิเคชัน Vue.js สามารถเรนเดอร์ที่ฝั่งเซิร์ฟเวอร์ได้ ซึ่งช่วยเพิ่มประสิทธิภาพในการโหลดหน้าเว็บและ SEO
2. **Static Site Generation (SSG)**: Nuxt.js รองรับการสร้างเว็บไซต์แบบสแตติกที่สามารถโฮสต์ได้ทุกที่
3. **Configuration over Convention**: Nuxt.js มีการตั้งค่าเริ่มต้นที่ดีและมีโครงสร้างที่ชัดเจน ทำให้นักพัฒนาสามารถเริ่มพัฒนาได้อย่างรวดเร็วโดยไม่ต้องกังวลเกี่ยวกับการตั้งค่ามากนัก

4. **Vue File Structure**: Nuxt.js ใช้ไฟล์และโฟลเดอร์โครงสร้างที่ชัดเจน ทำให้ง่ายต่อการจัดการและบำรุงรักษาโค้ด
5. **Middleware**: Nuxt.js มีระบบ middleware ที่ช่วยในการจัดการการรับส่งข้อมูลและการตรวจสอบสิทธิ์

การติดตั้งและเริ่มต้นใช้งาน Nuxt.js

การติดตั้ง Nuxt.js สามารถทำได้ผ่านการใช้เครื่องมือ CLI (Command Line Interface) โดยใช้คำสั่งดังนี้

```
npx create-nuxt-app <project-name>
```

คำสั่งนี้จะเปิด wizard ให้คุณเลือกการตั้งค่าต่าง ๆ เช่น:

- ชื่อโปรเจกต์
- ตัวเลือกสำหรับ UI framework เช่น Vuetify, Tailwind CSS
- การติดตั้งโมดูลเพิ่มเติมเช่น Axios, Content
- การตั้งค่าสำหรับการทดสอบและการจัดการสภาพแวดล้อม

โครงสร้างโปรเจกต์ Nuxt.js

เมื่อสร้างโปรเจกต์เสร็จแล้ว โครงสร้างของโปรเจกต์ Nuxt.js จะมีลักษณะดังนี้

```
├─ assets

├─ components

├─ layouts

├─ middleware

├─ pages
```

```
├ plugins
├ static
├ store
├ nuxt.config.js
└ package.json
```

- **assets**: โฟลเดอร์นี้ใช้เก็บไฟล์ที่ต้องการ preprocess เช่น SCSS, SASS, หรือ LESS
- **components**: ใช้เก็บ Vue components ที่จะนำมาใช้ในโปรเจกต์
- **layouts**: โฟลเดอร์นี้ใช้เก็บเลย์เอาต์ต่าง ๆ ของโปรเจกต์
- **middleware**: ใช้เก็บ middleware ของ Nuxt.js
- **pages**: ใช้เก็บไฟล์ Vue ที่จะเป็นหน้าเพจในโปรเจกต์
- **plugins**: ใช้เก็บปลั๊กอินของ Vue ที่ต้องการใช้งานในโปรเจกต์
- **static**: โฟลเดอร์นี้ใช้เก็บไฟล์ที่ต้องการให้เสิร์ฟตรง ๆ โดยไม่ต้อง preprocess เช่น ไฟล์รูปภาพหรือ favicon
- **store**: ใช้เก็บ Vuex store ของโปรเจกต์

การสร้างหน้าเพจใน Nuxt.js

ในการสร้างหน้าเพจใหม่ใน Nuxt.js เพียงแค่เพิ่มไฟล์ Vue ในโฟลเดอร์ Pages

```
// pages/index.vue

<template>

  <div>

    <h1>Welcome to My Nuxt.js App</h1>
```

```
  </div>

</template>



<script>

export default {

  name: 'HomePage'

}

</script>
```

Nuxt.js จะทำการกำหนดเส้นทาง (route) ให้โดยอัตโนมัติ โดย index.vue จะตรงกับเส้นทาง / และไฟล์อื่น ๆ ในโฟลเดอร์ pages จะมีเส้นทางที่สอดคล้องกับโครงสร้างของโฟลเดอร์นั้น ๆ

การใช้งาน Vuex ใน Nuxt.js

Nuxt.js รองรับการใช้งาน Vuex ในโฟลเดอร์ store โดยเพียงแค่สร้างไฟล์ index.js ภายในโฟลเดอร์ store

```
// store/index.js

export const state = () => ({

  counter: 0

})



export const mutations = {
```

```
  increment (state) {
    state.counter++
  }
}
```

จากนั้นใน component สามารถเข้าถึง store และ commit mutation ได้ดังนี้

```
<template>
  <div>
    <button @click="incrementCounter">Increment</button>
    <p>Counter: {{ counter }}</p>
  </div>
</template>

<script>
export default {
  computed: {
    counter() {
      return this.$store.state.counter
    }
```

```
  },
  methods: {
    incrementCounter() {
      this.$store.commit('increment')
    }
  }
}
</script>
```

การใช้งาน CSS Framework กับ Nuxt.js

Nuxt.js สามารถใช้งานร่วมกับ CSS framework ต่าง ๆ ได้ เช่น Tailwind CSS, Bootstrap, หรือ Bulma ในที่นี้จะแสดงตัวอย่างการติดตั้งและใช้งาน Tailwind CSS กับ Nuxt.js

การติดตั้ง Tailwind CSS

ติดตั้งแพ็กเกจที่จำเป็น

```
npm install @nuxtjs/tailwindcss tailwindcss@latest postcss@latest autoprefixer@latest
```

เพิ่มโมดูล Tailwind CSS ใน nuxt.config.js

```
// nuxt.config.js
export default {
  buildModules: [
    '@nuxtjs/tailwindcss'
  ]
```

```
}
```

สร้างไฟล์ tailwind.config.js

```
npx tailwindcss init
```

สร้างไฟล์ postcss.config.js

```
javascript
Copy code
// postcss.config.js
module.exports = {
  plugins: {
    tailwindcss: {},
    autoprefixer: {},
  },
}
```

เพิ่ม CSS ของ Tailwind ในไฟล์ assets/css/tailwind.css

```
/* assets/css/tailwind.css */
@tailwind base;
@tailwind components;
@tailwind utilities;
```

6. รวม CSS ของ Tailwind ใน nuxt.config.js:

```
javascript
Copy code
// nuxt.config.js
export default {
```

```
  css: [
    '@/assets/css/tailwind.css'
  ],
  buildModules: [
    '@nuxtjs/tailwindcss'
  ]
}
```

การใช้งาน Tailwind CSS

หลังจากการตั้งค่าเสร็จสิ้น สามารถใช้งาน Tailwind CSS ใน component ของ Nuxt.js ได้ทันที

```
<template>
  <div class="flex items-center justify-center min-h-screen bg-gray-100">
    <h1 class="text-4xl font-bold text-blue-500">Welcome to Nuxt.js with Tailwind CSS</h1>
  </div>
</template>
```

การตั้งค่าและการใช้งาน Nuxt.js
การตั้งค่าใน nuxt.config.js

ไฟล์ nuxt.config.js ใช้สำหรับการตั้งค่าต่าง ๆ ของ Nuxt.js เช่น การตั้งค่าโมดูล, การตั้งค่า build, และการกำหนดค่า plugins

```
export default {
  // Global page headers
  head: {
    title: 'my-nuxt-app',
    meta: [
      { charset: 'utf-8' },
      { name: 'viewport', content: 'width=device-width, initial-scale=1' },
      { hid: 'description', name: 'description', content: '' }
    ],
```

```
    link: [
      { rel: 'icon', type: 'image/x-icon', href: '/favicon.ico' }
    ]
  },

  // Global CSS
  css: [
    '@/assets/css/tailwind.css'
  ],

  // Plugins to run before rendering page
  plugins: [],

  // Auto import components
  components: true,

  // Modules for dev and build (recommended)
  buildModules: [
    '@nuxtjs/tailwindcss',
  ],

  // Modules
  modules: [
    '@nuxtjs/axios',
  ],

  // Build Configuration
  build: {}
}
```

การใช้งาน Plugins

Nuxt.js รองรับการใช้ปลั๊กอินเพื่อเพิ่มฟังก์ชันและความสามารถในโปรเจกต์ โดยปลั๊กอินสามารถกำหนดได้ใน plugins โฟลเดอร์

```
// plugins/my-plugin.js
import Vue from 'vue'
import MyPlugin from 'my-plugin'

Vue.use(MyPlugin)
```

และเพิ่มใน nuxt.config.js:

```
javascript
Copy code
export default {
  plugins: [
    '~/plugins/my-plugin.js'
  ]
}
```

Nuxt.js เป็นเฟรมเวิร์คที่มีประสิทธิภาพและง่ายต่อการใช้งานในการพัฒนาแอปพลิเคชัน Vue.js ด้วยการรองรับ SSR, SSG, และ SPA การใช้ Nuxt.js ช่วยลดภาระในการตั้งค่าและเพิ่มประสิทธิภาพในการพัฒนา ทำให้การสร้างเว็บแอปพลิเคชันมีความรวดเร็วและมีคุณภาพมากขึ้น

2.2.3 กรอบแนวคิด Fast API

ภาพที่ 1 Fast API

FastAPI เป็นเฟรมเวิร์คสำหรับการพัฒนา API ที่ออกแบบมาเพื่อความรวดเร็วและง่ายต่อการใช้งาน ด้วยการใช้ไวยากรณ์ของ Python และการทำงานที่รวดเร็ว ทำให้ FastAPI ได้รับความนิยมอย่างรวดเร็วในกลุ่มนักพัฒนา API

- **ประสิทธิภาพสูง**: FastAPI ถูกออกแบบมาเพื่อให้ทำงานได้เร็วมาก ใช้ประโยชน์จาก Starlette สำหรับ web framework และ Pydantic สำหรับ data validation ทำให้สามารถประมวลผลได้เร็วเทียบเท่ากับ Node.js และ Go
- **รองรับการพัฒนาแบบ Asynchronous**: สามารถใช้คำสั่ง async และ await เพื่อสร้าง asynchronous endpoints ซึ่งช่วยให้สามารถจัดการกับการร้องขอหลายๆครั้งพร้อมกันได้อย่างมีประสิทธิภาพ
- **การตรวจสอบและตรวจสอบความถูกต้องของข้อมูลอัตโนมัติ**: ด้วยการใช้ Pydantic ทำให้สามารถตรวจสอบและ validate ข้อมูลที่รับเข้ามาและส่งออกไปได้อย่างง่ายดาย
- **เอกสาร API อัตโนมัติ**: FastAPI จะสร้างเอกสาร API แบบ interactive โดยอัตโนมัติผ่าน OpenAPI และ Swagger UI ทำให้การทดสอบและการใช้งาน API ง่ายขึ้นมาก
- **ใช้งานง่าย**: ไวยากรณ์ของ FastAPI คล้ายกับ Flask ทำให้ผู้ที่เคยใช้ Flask สามารถเริ่มต้นใช้งาน FastAPI ได้อย่างรวดเร็ว
- **รองรับการสร้างและจัดการ Background Tasks**: สามารถสร้างและจัดการ background tasks ได้ง่ายๆ ด้วยการใช้คำสั่ง async

FastAPI เป็นเฟรมเวิร์คที่ทรงพลังสำหรับการพัฒนา API ที่รวดเร็ว มีประสิทธิภาพ และง่ายต่อการใช้งาน ด้วยคุณสมบัติที่หลากหลายและการทำงานที่รวดเร็ว ทำให้ FastAPI เป็นทางเลือกที่ดีสำหรับนักพัฒนาที่ต้องการสร้าง API ที่ทันสมัยและมีประสิทธิภาพ

2.2.4 DrizzleORM

ภาพที่ 2 DrizzleORM

DrizzleORM เป็น Object-Relational Mapper (ORM) ที่ช่วยให้การจัดการฐานข้อมูลเป็นเรื่องง่ายและมีประสิทธิภาพสำหรับนักพัฒนาเว็บและแอปพลิเคชันโมเดิร์น มีจุดเด่นในการทำงานร่วมกับ TypeScript และ JavaScript และถูกออกแบบมาให้มีความเบาและง่ายต่อการใช้งานจุดเด่นของ DrizzleORM สามารถสรุปได้ดังนี้

- ใช้งานง่ายและเบา: DrizzleORM ถูกออกแบบมาให้ใช้งานง่าย มีขนาดเบา และไม่มีการซ่อนกระบวนการซับซ้อนภายใน ช่วยให้นักพัฒนาสามารถเข้าใจและควบคุมการทำงานของฐานข้อมูลได้อย่างเต็มที่
- รองรับ TypeScript: DrizzleORM ทำงานได้ดีกับ TypeScript ช่วยให้นักพัฒนาที่ใช้ TypeScript สามารถใช้งานได้อย่างสะดวกและปลอดภัยมากขึ้นด้วยการตรวจสอบประเภทข้อมูลในเวลาเขียนโค้ด
- การสร้างและการจัดการตารางที่ง่าย: DrizzleORM ช่วยให้การสร้างและการจัดการตารางฐานข้อมูลเป็นเรื่องง่าย โดยใช้โครงสร้างที่ชัดเจนและเข้าใจง่าย
- การสร้าง Query ที่มีประสิทธิภาพ: ช่วยให้นักพัฒนาสามารถสร้าง Query ที่ซับซ้อนและมีประสิทธิภาพได้ง่ายๆ ด้วยการใช้งาน API ที่เป็นมิตรและมีประสิทธิภาพ
- รองรับการเชื่อมต่อฐานข้อมูลหลายประเภท: สามารถใช้งานกับฐานข้อมูลหลากหลายประเภท เช่น PostgreSQL, MySQL, SQLite เป็นต้น

2.2.5 KTB FastPay

KTB Fast Pay เป็นบริการชำระเงินผ่านระบบดิจิทัลของธนาคารกรุงไทย (Krung Thai Bank - KTB) ที่ถูกออกแบบมาเพื่อตอบสนองความต้องการของลูกค้าที่ต้องการความสะดวก รวดเร็ว และปลอดภัยในการทำธุรกรรมทางการเงินออนไลน์ ไม่ว่าจะเป็นการชำระค่าสินค้าและบริการ การโอนเงิน หรือการจ่ายบิลต่างๆ

- **ความรวดเร็วในการทำธุรกรรม**: ระบบ KTB Fast Pay ถูกออกแบบมาเพื่อให้การทำธุรกรรมเสร็จสิ้นได้ภายในไม่กี่วินาที ช่วยลดระยะเวลาในการรอคอย
- **ความปลอดภัยสูง**: ใช้เทคโนโลยีการเข้ารหัสและการตรวจสอบตัวตนที่มีความปลอดภัยสูง เพื่อปกป้องข้อมูลส่วนตัวและข้อมูลการเงินของลูกค้า
- **การใช้งานที่ง่ายดาย**: มีอินเทอร์เฟซที่ใช้งานง่าย ทำให้ลูกค้าสามารถทำธุรกรรมได้โดยไม่ยุ่งยาก
- **รองรับหลายแพลตฟอร์ม**: สามารถใช้งานได้ทั้งบนเว็บไซต์และแอปพลิเคชันมือถือ ทำให้ลูกค้าสามารถเข้าถึงบริการได้ทุกที่ทุกเวลา
- **การแจ้งเตือนทันที**: มีระบบการแจ้งเตือนผ่าน SMS และอีเมลเมื่อทำธุรกรรมสำเร็จ เพื่อให้ลูกค้ารับรู้และตรวจสอบได้ทันที

วิธีการใช้งาน KTB Fast Pay

- **ลงทะเบียนและเข้าสู่ระบบ**: ลูกค้าต้องลงทะเบียนบัญชี KTB Fast Pay และเข้าสู่ระบบผ่านแอปพลิเคชันหรือเว็บไซต์ของธนาคารกรุงไทย
- **เลือกประเภทของธุรกรรม**: ลูกค้าสามารถเลือกประเภทของธุรกรรมที่ต้องการทำ เช่น การโอนเงิน การจ่ายบิล หรือการชำระค่าสินค้าและบริการ
- **กรอกข้อมูลธุรกรรม**: กรอกข้อมูลที่จำเป็นสำหรับธุรกรรมนั้นๆ เช่น หมายเลขบัญชีปลายทาง จำนวนเงิน และรายละเอียดอื่นๆ
- **ยืนยันและดำเนินการ**: ตรวจสอบข้อมูลและยืนยันการทำธุรกรรม ระบบจะดำเนินการและแจ้งเตือนลูกค้าเมื่อธุรกรรมเสร็จสิ้น

KTB Fast Pay เป็นบริการชำระเงินที่รวดเร็วและปลอดภัยที่ธนาคารกรุงไทยนำเสนอเพื่อตอบสนองความต้องการของลูกค้าในยุคดิจิทัล ด้วยคุณสมบัติที่หลากหลายและการใช้งานที่ง่ายดาย ทำให้ลูกค้าสามารถทำธุรกรรมได้สะดวกและปลอดภัยมากยิ่งขึ้น ไม่ว่าจะเป็นการชำระค่าสินค้าและบริการ การโอนเงิน หรือการจ่ายบิลต่างๆ ระบบนี้ยังมีการแจ้งเตือนทันทีเพื่อให้ลูกค้ารับรู้และตรวจสอบสถานะของธุรกรรมได้ตลอดเวลา

บทที่ 3

ขั้นตอนวิธีและการดำเนินงานพัฒนา

AS-IS Model: การชำระเงินแบบเดิม (โอนผ่านธนาคารและส่งสลิปทางอีเมล์)

- ขั้นตอนที่ 1: ผู้ใช้ทำการลงทะเบียน
  กรอกข้อมูลลงทะเบียน: ผู้ใช้กรอกข้อมูลส่วนตัวและรายละเอียดการลงทะเบียนผ่านหน้าเว็บไซต์หรือแบบฟอร์มออนไลน์
- ยืนยันการลงทะเบียน: ผู้ใช้ได้รับการยืนยันการลงทะเบียนทางอีเมล์หรือผ่านหน้าเว็บไซต์ พร้อมรายละเอียดบัญชีธนาคารสำหรับการโอนเงิน
- ขั้นตอนที่ 2: การชำระเงิน
  โอนเงินผ่านธนาคาร: ผู้ใช้ทำการโอนเงินผ่านธนาคาร (Internet Banking, Mobile Banking, ATM หรือที่สาขา) เก็บสลิปการโอน ซึ่งผู้ใช้เก็บสลิปการโอนเงินเป็นหลักฐานการชำระเงิน
- ขั้นตอนที่ 3: การส่งสลิปการชำระเงิน
  ส่งสลิปทางอีเมล์: ผู้ใช้ถ่ายรูปหรือสแกนสลิปการโอนเงินแล้วส่งอีเมล์ไปยังทีมงาน/ผู้จัดการประชุมพร้อมกับรายละเอียดการลงทะเบียน
  ยืนยันการรับสลิป: ทีมงานตรวจสอบสลิปการชำระเงินและยืนยันการรับเงินผ่านอีเมล์หรือระบบ
- ขั้นตอนที่ 4: การออกใบเสร็จรับเงิน
  ออกใบเสร็จรับเงิน: ทีมงานตรวจสอบข้อมูลการชำระเงินและออกใบเสร็จรับเงินให้ผู้ใช้
  ส่งใบเสร็จรับเงิน: ใบเสร็จรับเงินถูกส่งไปยังผู้ใช้ทางอีเมล์หรือในรูปแบบกระดาษ

To-Be Model: การชำระเงินแบบใหม่ (ผ่านบัตรเครดิต/เดบิต พร้อมออกใบเสร็จรับเงินอัตโนมัติ)

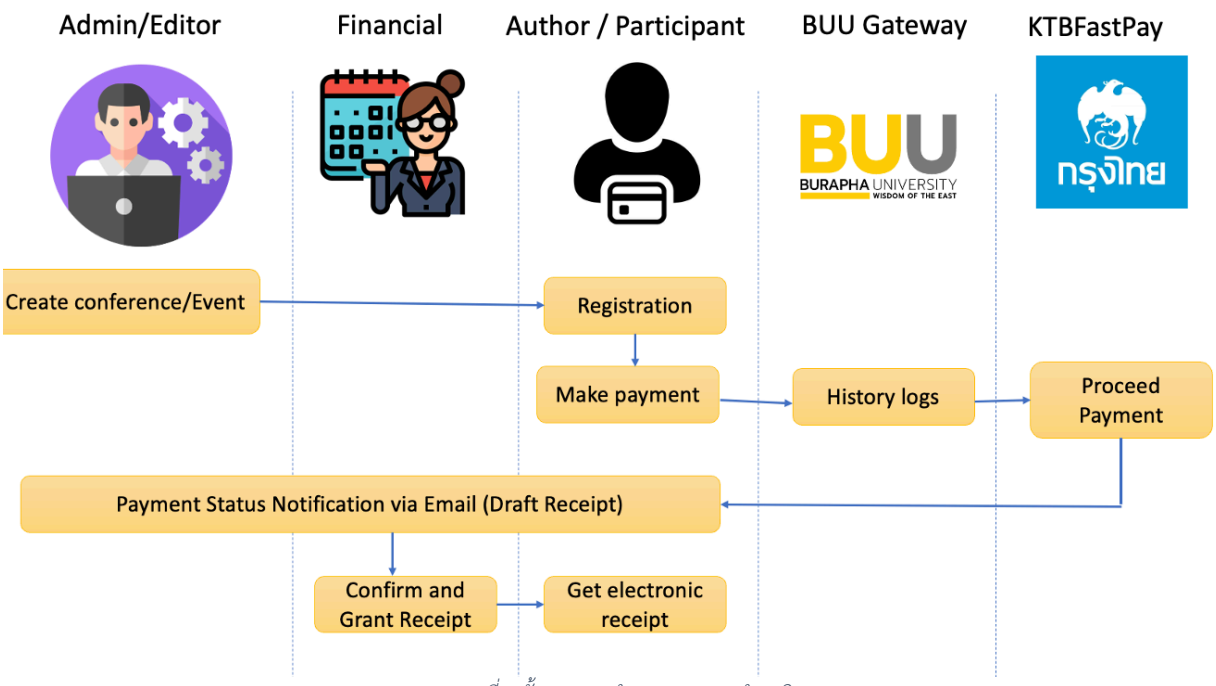
- ขั้นตอนที่ 1: ผู้ใช้ทำการลงทะเบียน
  กรอกข้อมูลลงทะเบียน: ผู้ใช้กรอกข้อมูลส่วนตัวและรายละเอียดการลงทะเบียนผ่านหน้าเว็บไซต์หรือแบบฟอร์มออนไลน์
  ยืนยันการลงทะเบียน: ผู้ใช้ได้รับการยืนยันการลงทะเบียนทางอีเมล์หรือผ่านหน้าเว็บไซต์ พร้อมข้อมูลสำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต
- ขั้นตอนที่ 2: การชำระเงิน

เลือกวิธีการชำระเงิน: ผู้ใช้เลือกชำระเงินผ่านบัตรเครดิต/เดบิตในหน้าเว็บไซต์หรือแบบฟอร์มออนไลน์

- กรอกข้อมูลบัตรเครดิต/เดบิต: ผู้ใช้กรอกข้อมูลบัตรเครดิต/เดบิต (หมายเลขบัตร, วันหมดอายุ, CVV) ในหน้าเว็บไซต์ที่ปลอดภัย
  ยืนยันการชำระเงิน: ระบบทำการประมวลผลการชำระเงินผ่าน Payment Gateway (เช่น Stripe, PayPal) และยืนยันการชำระเงิน
- ขั้นตอนที่ 3: การออกใบเสร็จรับเงินอัตโนมัติ
  ออกใบเสร็จรับเงิน: ระบบทำการตรวจสอบการชำระเงินและออกใบเสร็จรับเงินอัตโนมัติในรูปแบบอิเล็กทรอนิกส์
  ส่งใบเสร็จรับเงิน: ใบเสร็จรับเงินถูกส่งไปยังผู้ใช้ทางอีเมล์โดยอัตโนมัติ

ความแตกต่างระหว่าง AS-IS Model และ To-Be Model

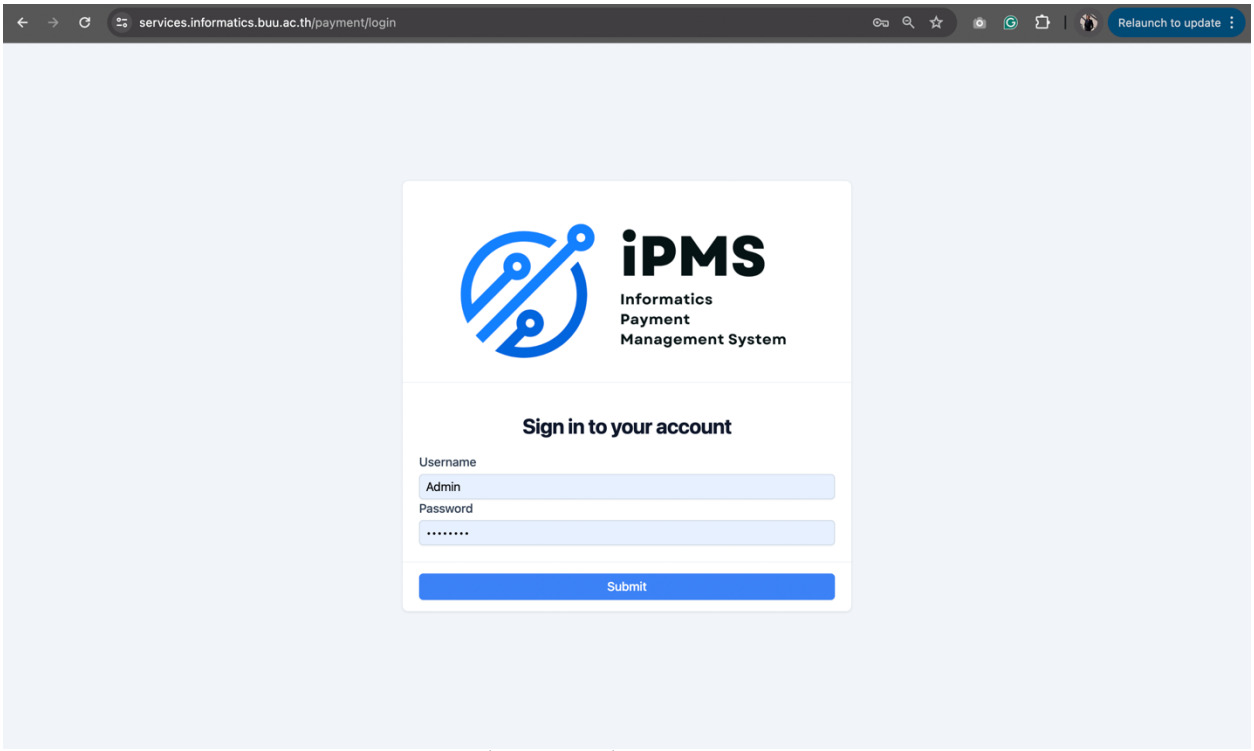
- ความสะดวกสบาย: ระบบใหม่ช่วยลดขั้นตอนที่ยุ่งยากในการโอนเงินและการส่งสลิปทางอีเมล์ ผู้ใช้สามารถทำรายการได้ทั้งหมดในหน้าเว็บไซต์
- ความรวดเร็ว: การชำระเงินและการออกใบเสร็จรับเงินเกิดขึ้นอย่างรวดเร็วและอัตโนมัติ ลดเวลาที่ใช้ในการตรวจสอบและยืนยันการชำระเงิน
- ความปลอดภัย: การใช้ Payment Gateway ช่วยเพิ่มความปลอดภัยในการทำธุรกรรมทางการเงิน ลดความเสี่ยงจากการใช้ข้อมูลบัตรเครดิต/เดบิต
- การลดข้อผิดพลาด: การประมวลผลอัตโนมัติช่วยลดข้อผิดพลาดที่อาจเกิดขึ้นจากการตรวจสอบและการบันทึกข้อมูลด้วยมือ
- ประสบการณ์ของผู้ใช้: ผู้ใช้ได้รับประสบการณ์ที่ดีขึ้นจากการใช้ระบบที่มีความรวดเร็วและสะดวกสบายมากขึ้น



ภาพที่ 3 ขั้นตอนการทำงานของระบบชำระเงิน

ในการพัฒนาระบบสำหรับชำระเงินค่าลงทะเบียนงานประชุมวิชาการ ผู้ดูแลระบบสามารถ Log in เพื่อทำการสร้างและกำหนดค่าต่าง ๆ ให้กับงานประชุมวิชาการ และสามารถสร้าง URL เพื่อใช้ในการลงทะเบียน เมื่อผู้เข้าร่วมหรือผู้เขียนบทความทำการลงทะเบียน จะสามมรถชำระเงินผ่านการโอนเงินผ่านธนาคารและชำระผ่านบัตรเครดิตหรือเดบิตได้ จากนั้นระบบ ฯ จะดำเนินการสร้างใบเสร็จรับเงินชั่วคราวและส่งผ่าน Email โดยอัตโนมัติ จากนั้นเจ้าหน้าที่การเงินสามารถตรวจสอบยอดชำระและดำเนินการออกใบเสร็จรับเงินทางการได้

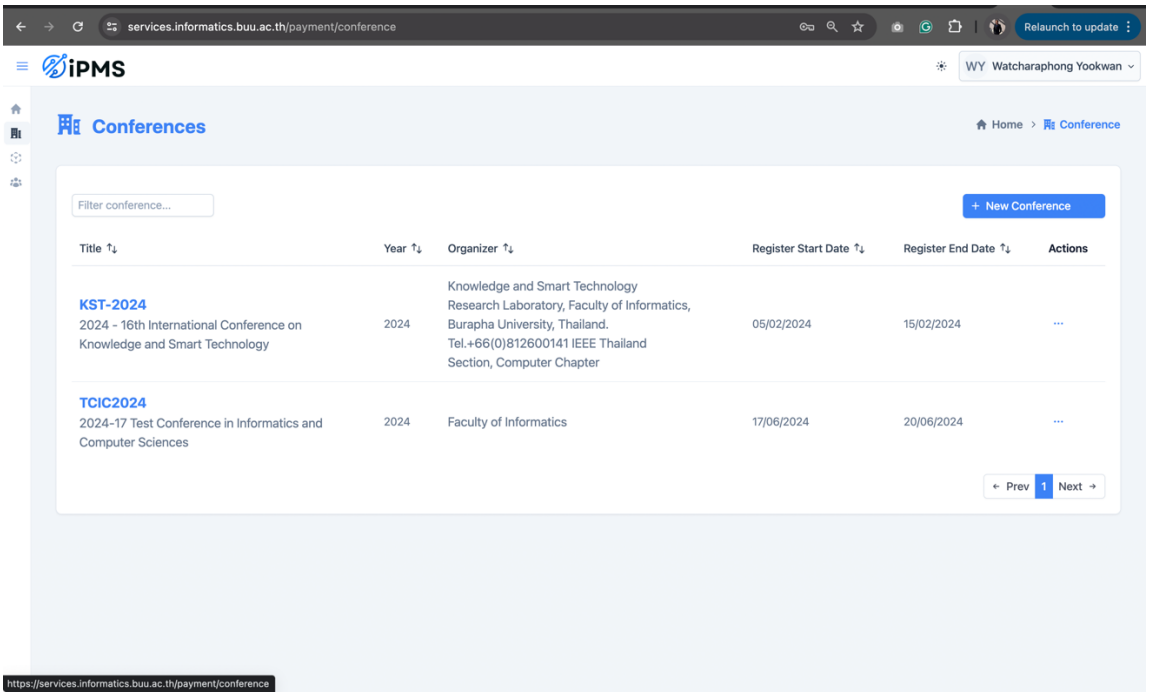
3.1 หน้าเว็บลงชื่อเข้าใช้สำหรับผู้ดูแลระบบ



ภาพที่ 4 หน้าเว็บลงชื่อเข้าใช้สำหรับผู้ดูแลระบบ

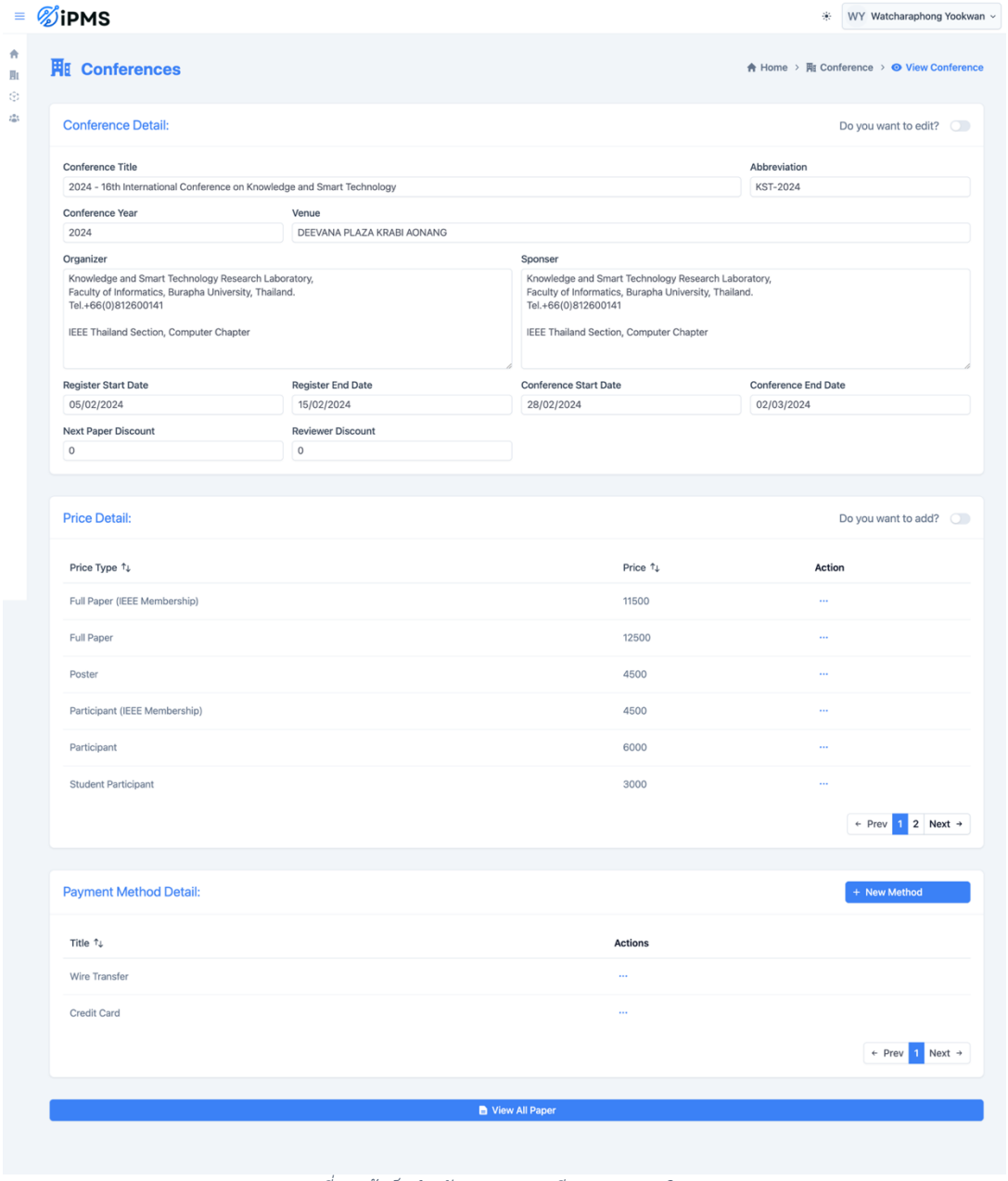
ในหน้าเว็บสำหรับลงชื่อเข้าใช้ จะไม่มีการให้ผู้ใช้สมัครใช้บริการ เนื่องจากผู้ใช้จะต้องถูกสร้างโดยผู้ดูแลเท่านั้นจึงจะสามารถเข้าใช้งานได้

3.2 หน้าเว็บแสดงรายการงานประชุมวิชาการ (Conference) หรืองานกิจกรรมที่กำลังใช้งานอยู่



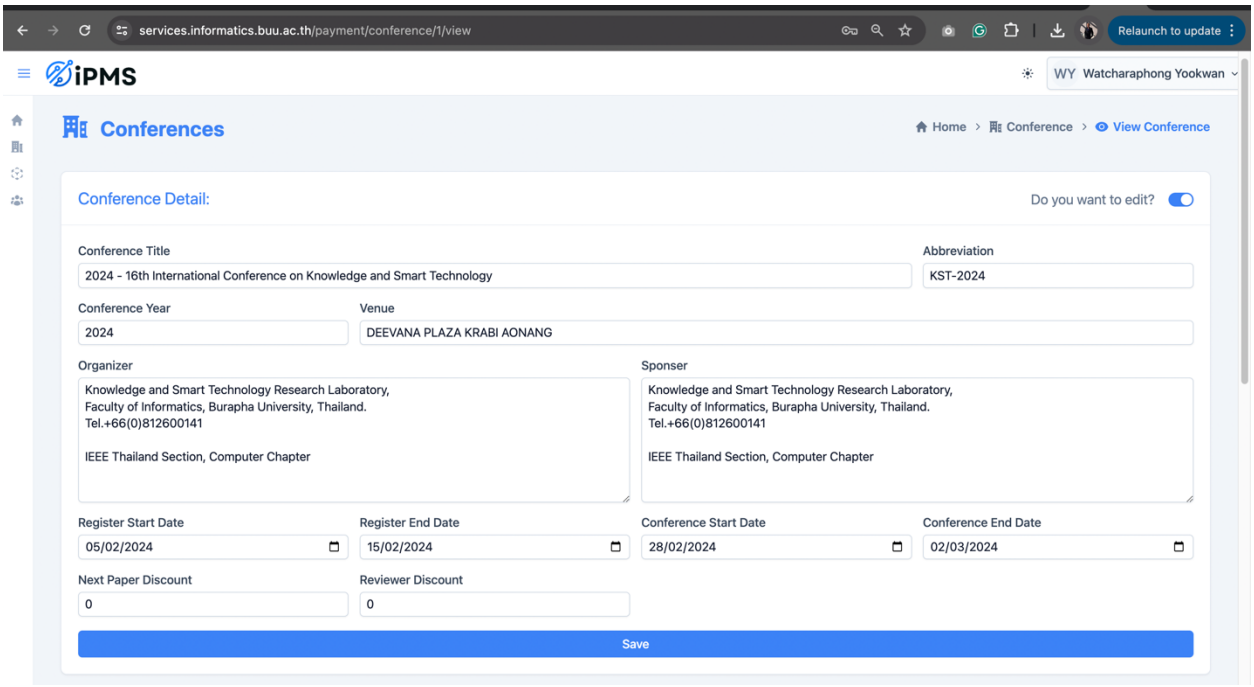
ภาพที่ 5 หน้าเว็บแสดงรายการงานประชุมวิชาการ (Conference) หรืองานกิจกรรมที่กำลังใช้งานอยู่

3.3 หน้าเว็บสำหรับแสดงรายละเอียดงานประชุมวิชาการ



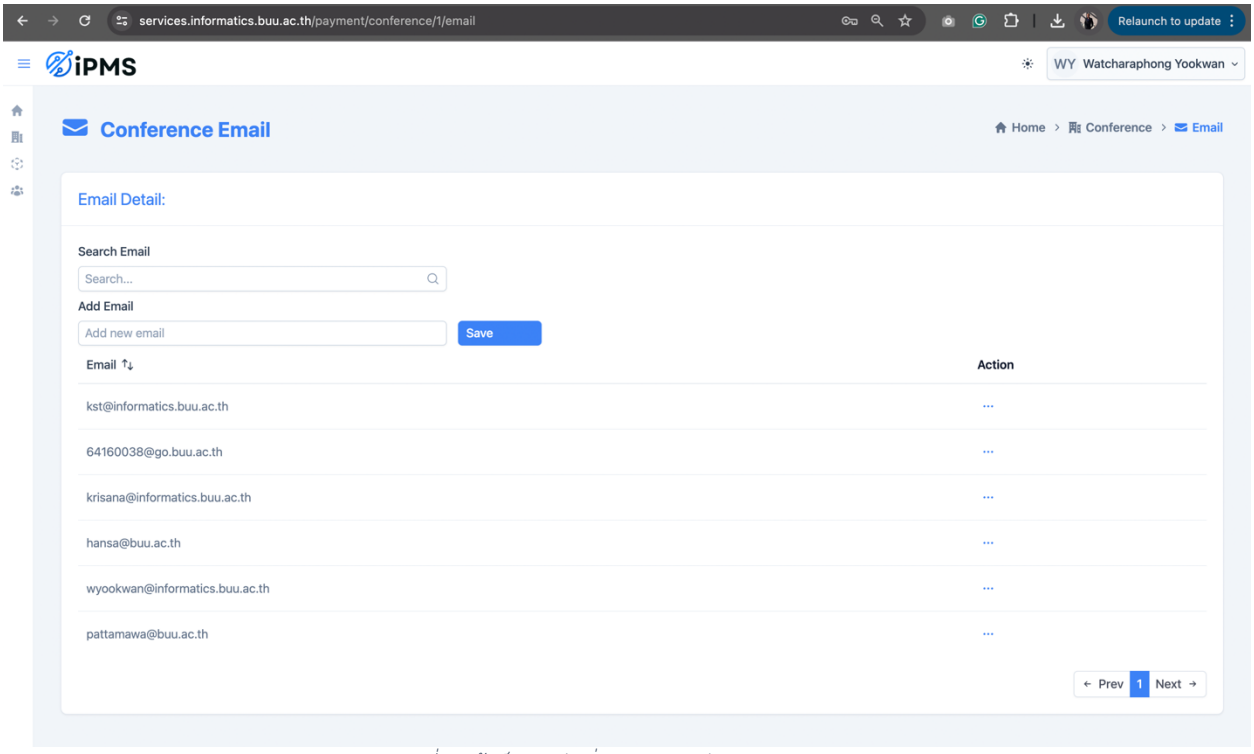
ภาพที่ 6 หน้าเว็บสำหรับแสดงรายละเอียดงานประชุมวิชาการ

3.4 หน้าเว็บสำหรับแก้ไขรายละเอียดงานประชุมวิชาการ



ภาพที่ 7 หน้าเว็บสำหรับแก้ไขรายละเอียดงานประชุมวิชาการ

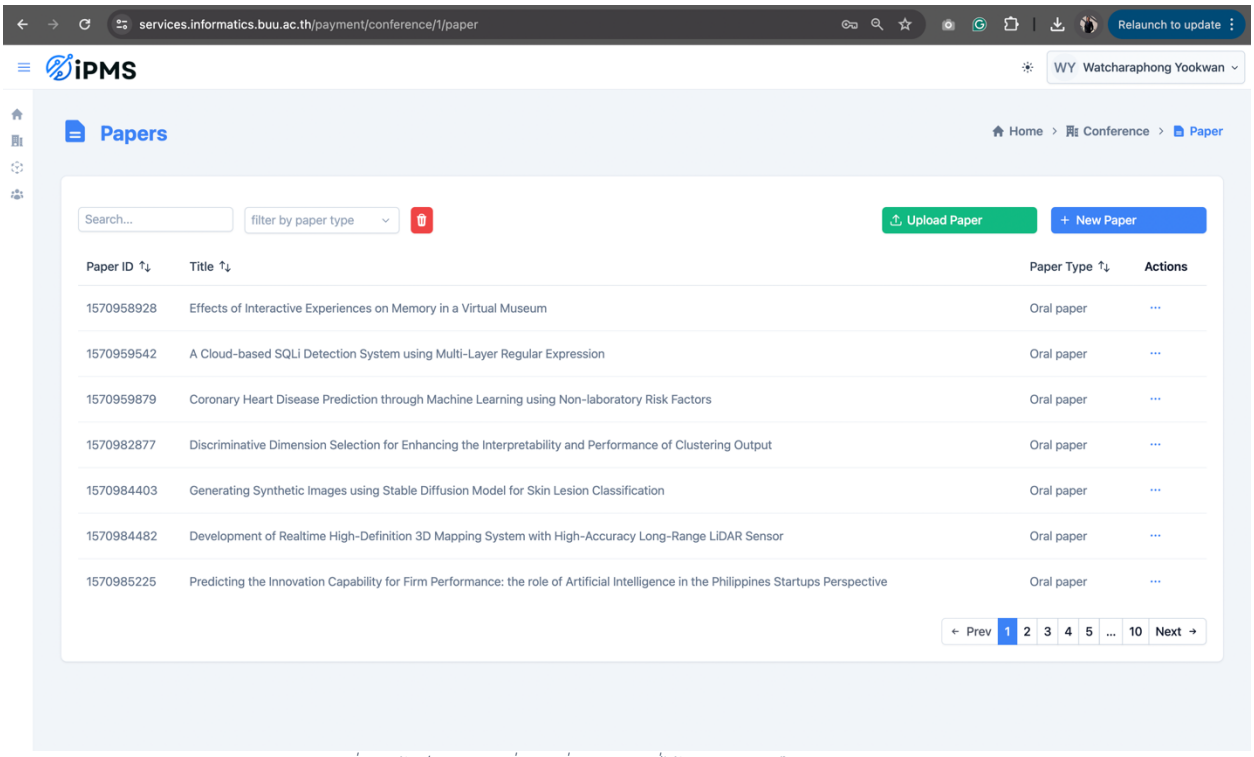
3.5 หน้าเว็บสำหรับเพิ่ม E-mail สำหรับงานประชุมวิชาการ



ภาพที่ 8 หน้าเว็บสำหรับเพิ่ม E-mail สำหรับงานประชุมวิชาการ

ผู้ดูแลสามารถเพิ่มที่อยู่ E-mail สำหรับส่งออก และสามารถเพิ่ม E-mail list สำหรับกำหนดผู้รับที่จะส่งถึงในแต่ละครั้งที่ส่งได้ (CC)

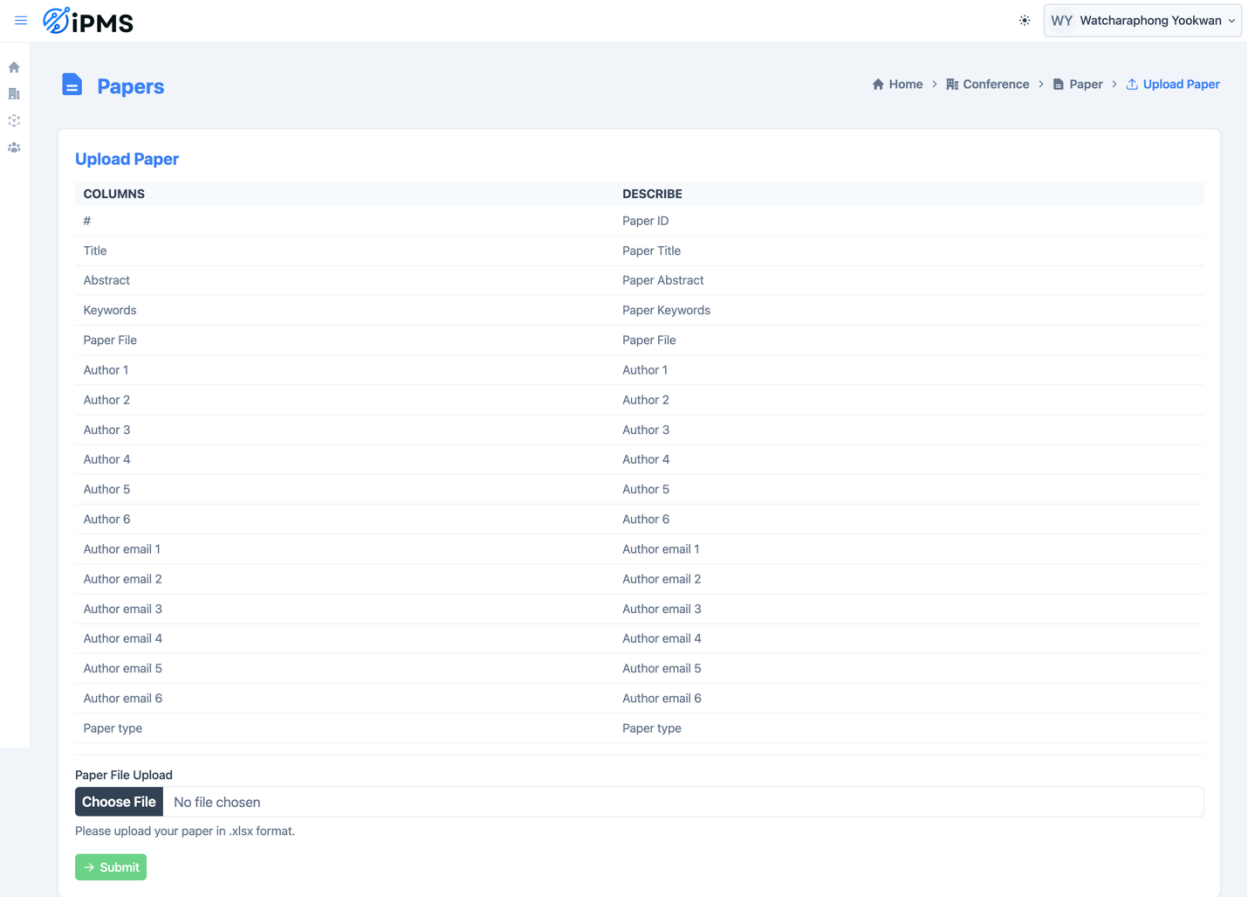
3.6 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการ



ภาพที่ 9 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการ

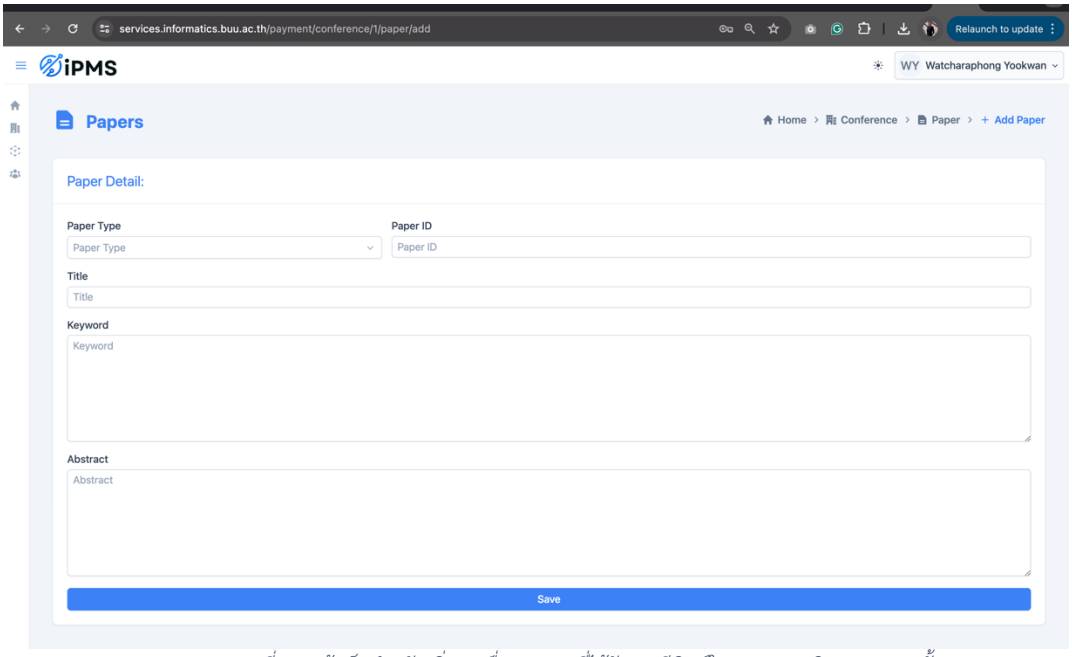
โดยสามารถเพิ่มรายชื่อบทความที่ถูกเก็บอยู่ในไฟล์ Spreadsheet ตามรูปแบบที่กำหนดได้และยังสามารถเพิ่มรายชื่อแบบทีละบทความได้

3.7 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการรูปแบบ Spreadsheet



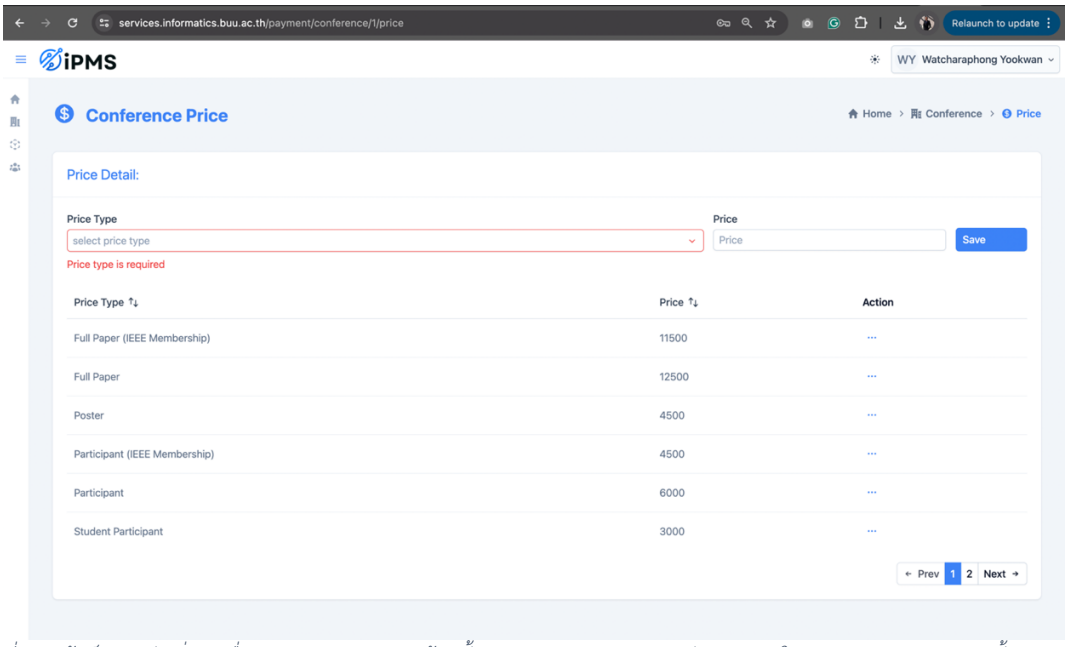
ภาพที่ 10 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการรูปแบบ Spreadsheet

3.8 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการแบบครั้งละบทความ



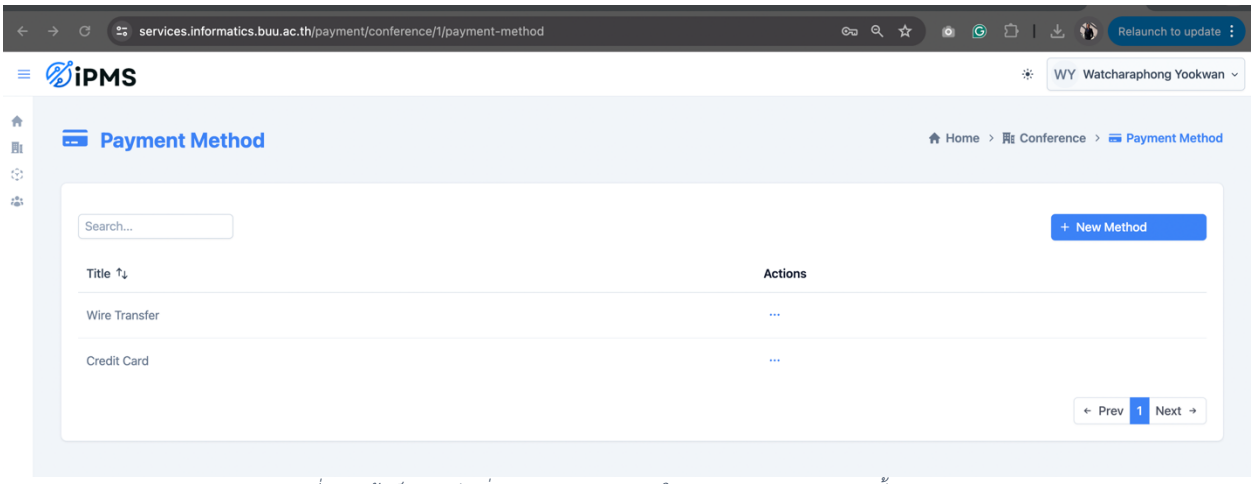
ภาพที่ 11 หน้าเว็บสำหรับเพิ่มรายชื่อบทความที่ได้รับการตีพิมพ์ในงานประชุมวิชาการแบบครั้งละบทความ

3.9 หน้าเว็บสำหรับเพิ่มรายชื่อประเภทของบทความพร้อมทั้งกำหนดราคาของบทความแต่ละประเภทในงานประชุมวิชาการแบบครั้งละบทความ



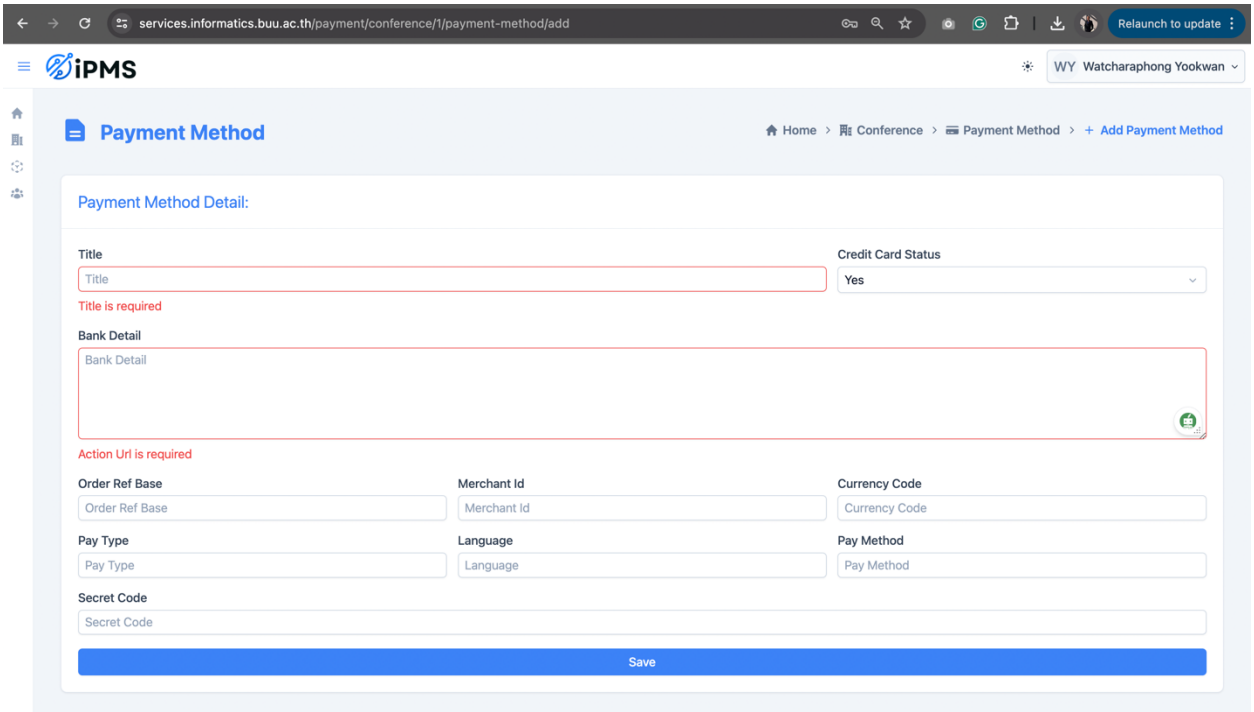
ภาพที่ 12 หน้าเว็บสำหรับเพิ่มรายชื่อประเภทของบทความพร้อมทั้งกำหนดราคาของบทความแต่ละประเภทในงานประชุมวิชาการแบบครั้งละบทความ

3.10 หน้าเว็บสำหรับเพิ่มประเภทการชำระเงินในงานประชุมวิชาการแบบครั้งละบทความ



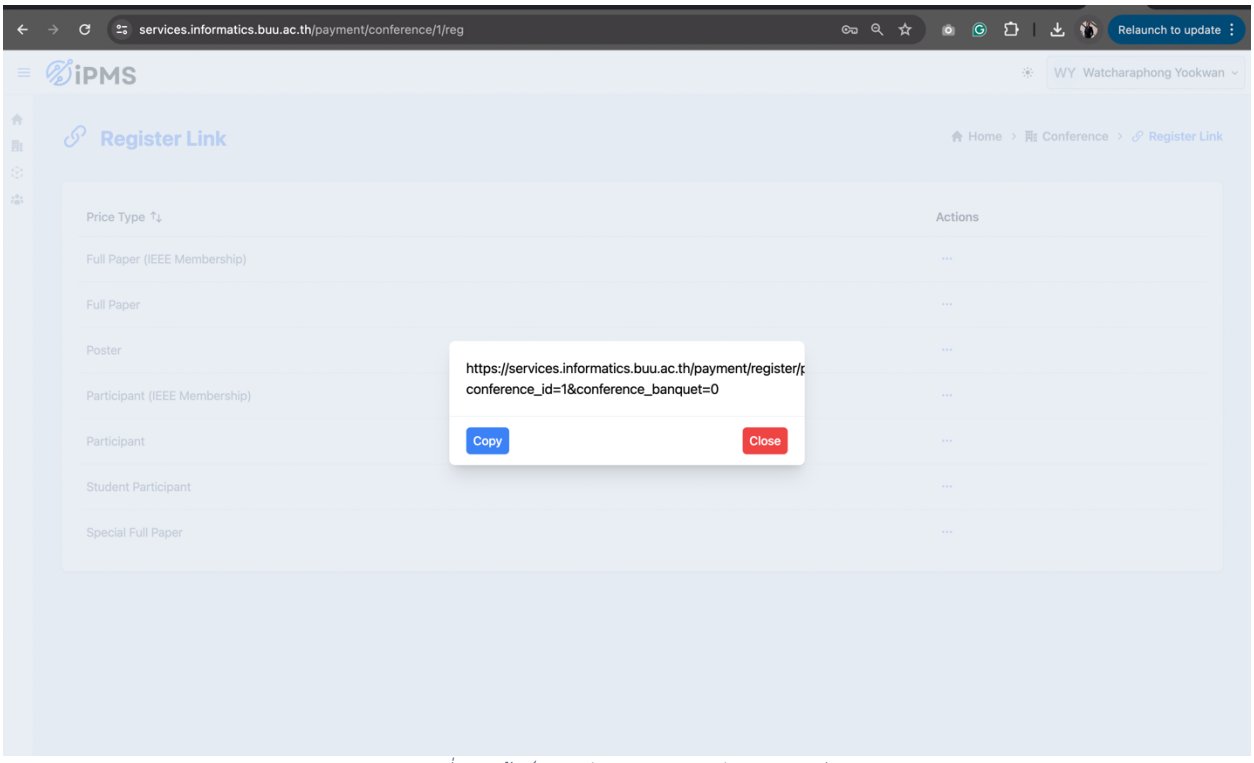
ภาพที่ 13 หน้าเว็บสำหรับเพิ่มประเภทการชำระเงินในงานประชุมวิชาการแบบครั้งละบทความ

3.11 หน้าเว็บสำหรับกำหนดค่า Parameter สำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต



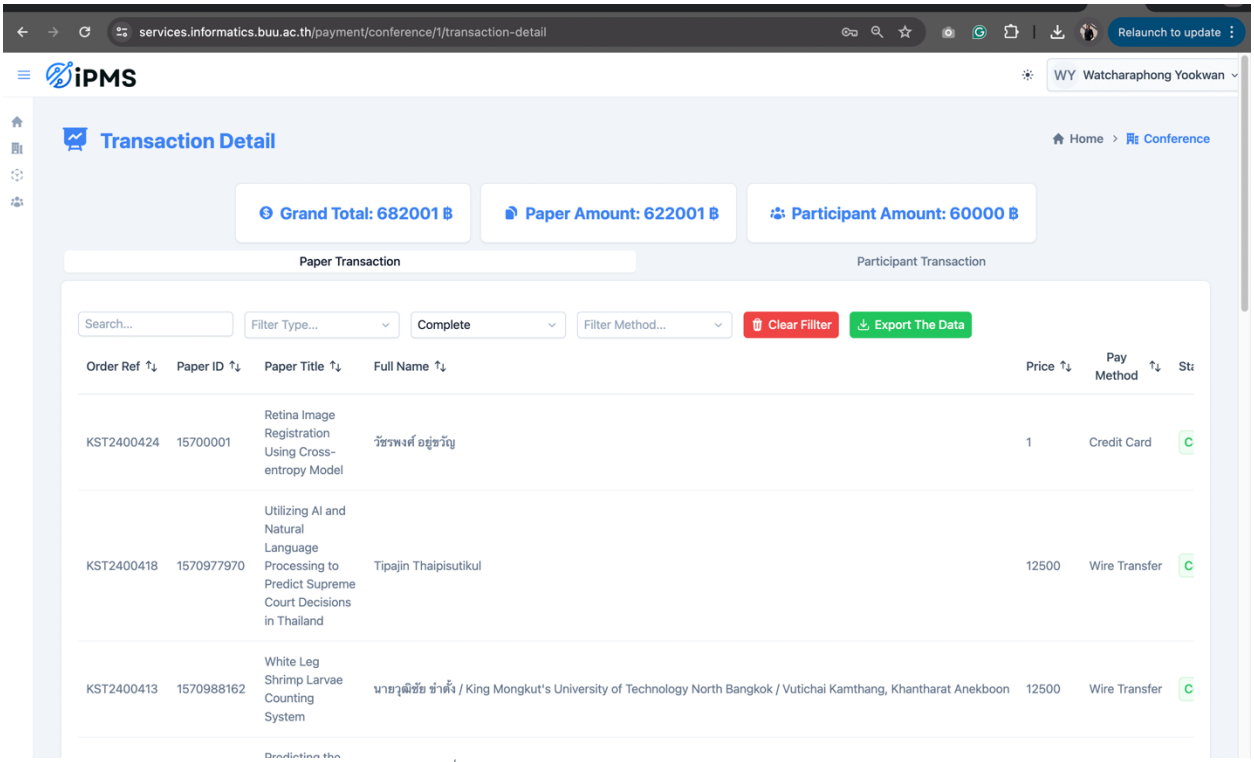
ภาพที่ 14 หน้าเว็บสำหรับกำหนดค่า Parameter สำหรับการชำระเงินผ่านบัตรเครดิต/เดบิต

3.12 หน้าเว็บสำหรับเสดง URL สำหรับการลงทะเบียน



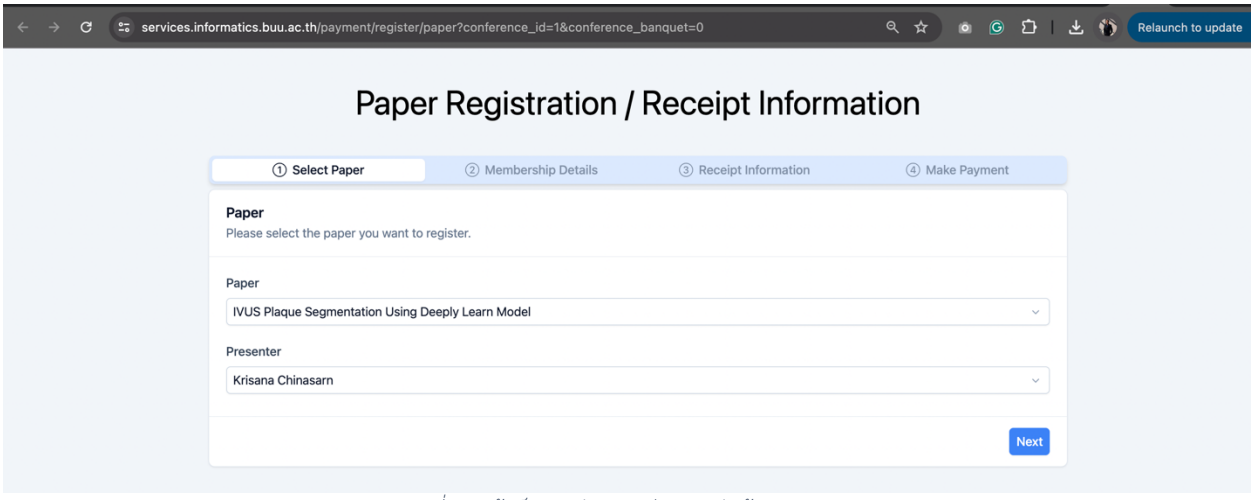
ภาพที่ 15 หน้าเว็บสำหรับเสดง URL สำหรับการลงทะเบียน

3.13 หน้าเว็บสำหรับแสดง Transactions การชำระเงิน



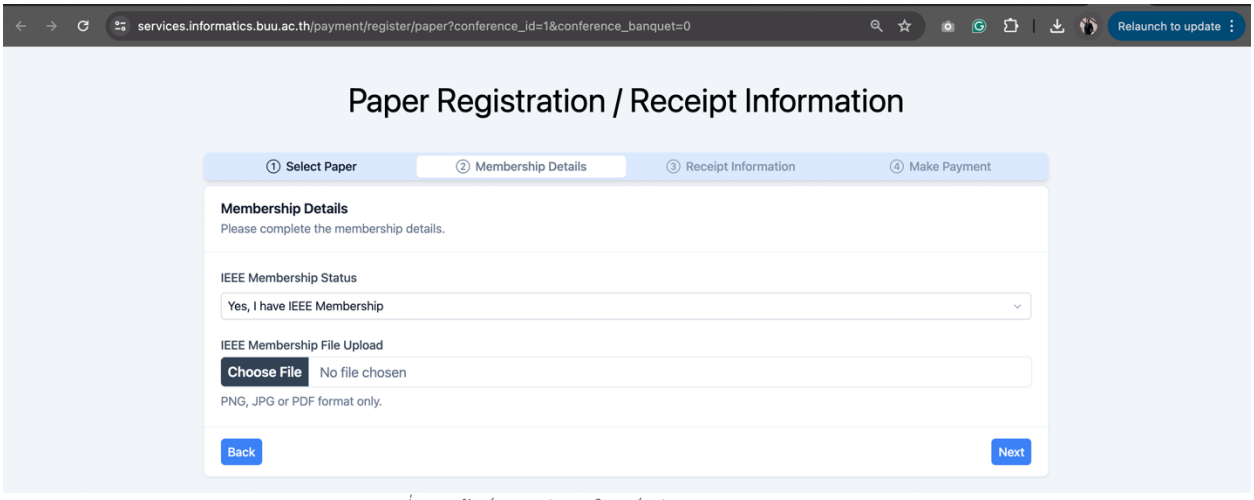
ภาพที่ 16 หน้าเว็บสำหรับแสดง Transactions การชำระเงิน

3.14 หน้าเว็บสำหรับลงทะเบียนสำหรับเจ้าของบทความ



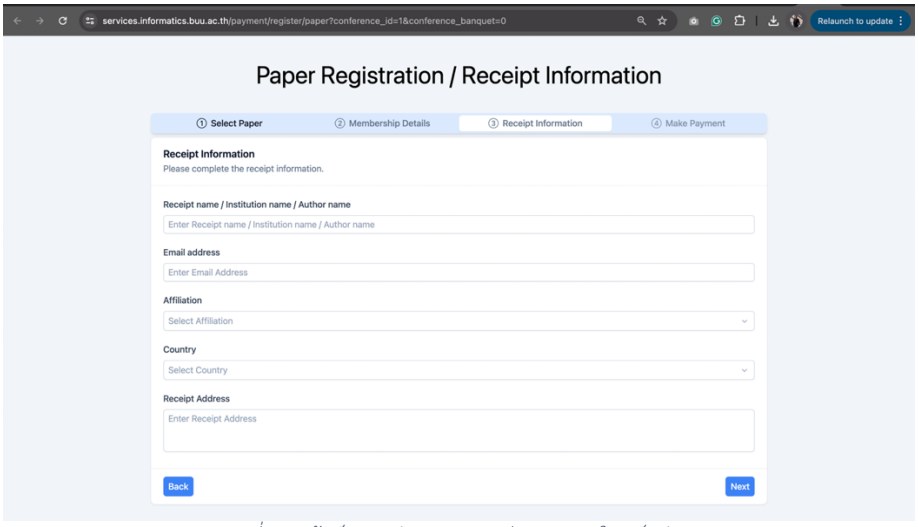
ภาพที่ 17 หน้าเว็บสำหรับลงทะเบียนสำหรับเจ้าของบทความ

โดยผู้ลงทะเบียนเพื่อเข้าร่วมงานจะต้องเลือกบทความที่ตนเองเป็นผู้เขียนและเลือกรายชื่อของผู้นำเสนอ (Presenter) จากนั้นผู้เขียนจะต้องเลือกว่าเป็นสมาชิกของ IEEE หรือไม่ (ในกรณีที่มีการกำหนดว่างานประชุมวิชาการนี้สนับสนุนโดยสมาคมหรือไม่) และเข้าสู่หน้าเว็บสำหรับออกใบเสร็จรับเงินต่อไป



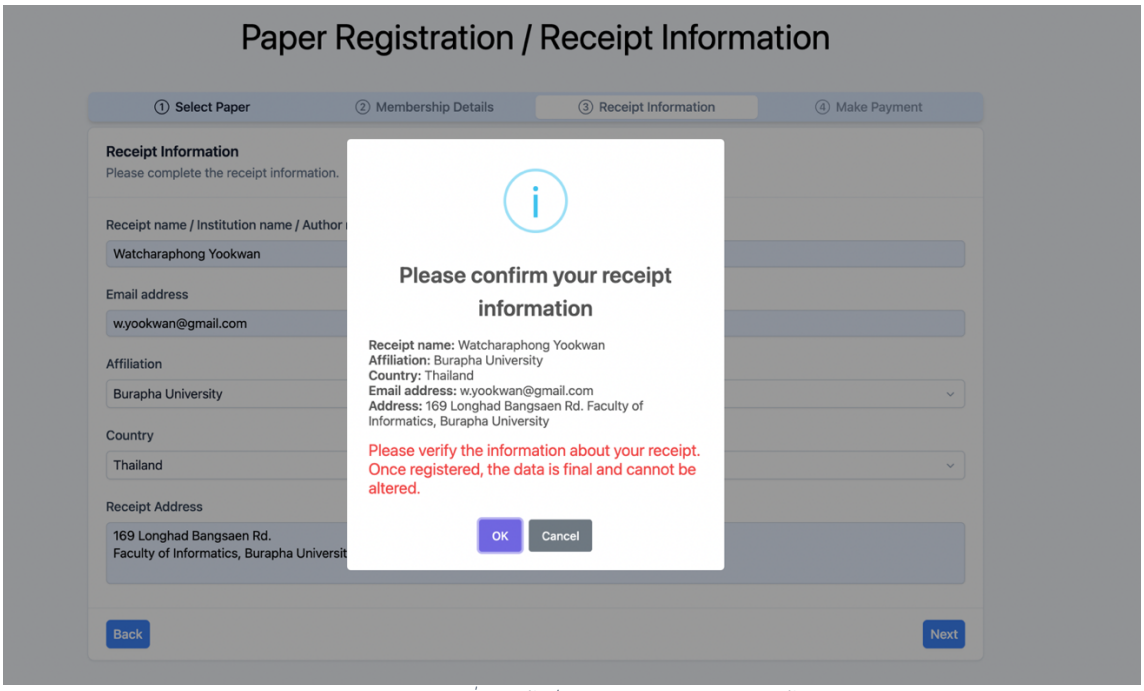
ภาพที่ 18 หน้าเว็บสำหรับออกใบเสร็จรับเงิน

3.15 หน้าเว็บสำหรับระบุรายละเอียดการออกใบเสร็จรับเงิน



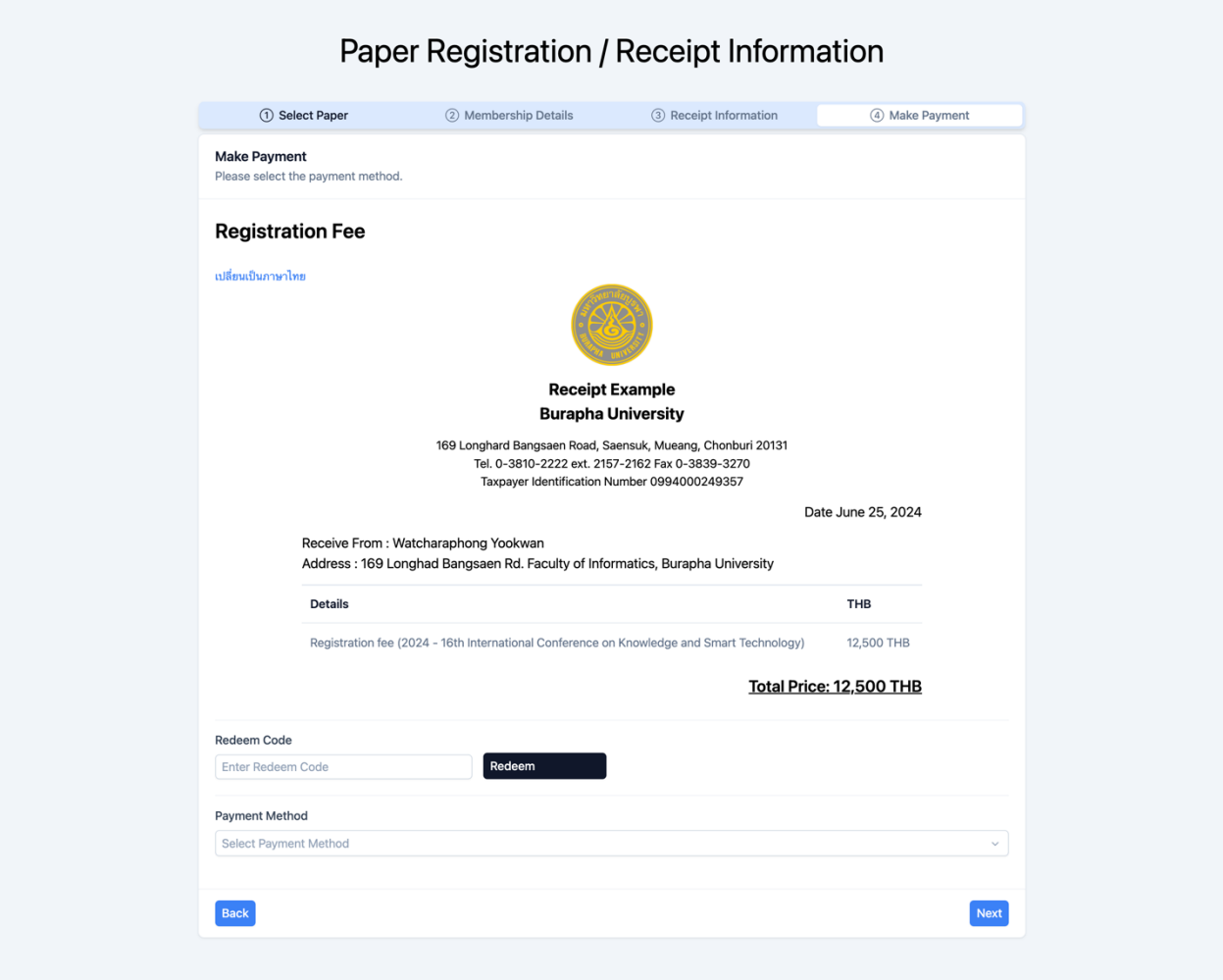
ภาพที่ 19 หน้าเว็บสำหรับระบุรายละเอียดการออกใบเสร็จรับเงิน

เมื่อระบุรายละเอียดครบถ้วนแล้ว ผู้ลงทะเบียนจะต้องตรวจสอบและยืนยันข้อมูลก่อนจึงสามารถดำเนินการในขั้นถัดไปได้



ภาพที่ 20 หน้าเว็บสำหรับตรวจสอบและยืนยันข้อมูล

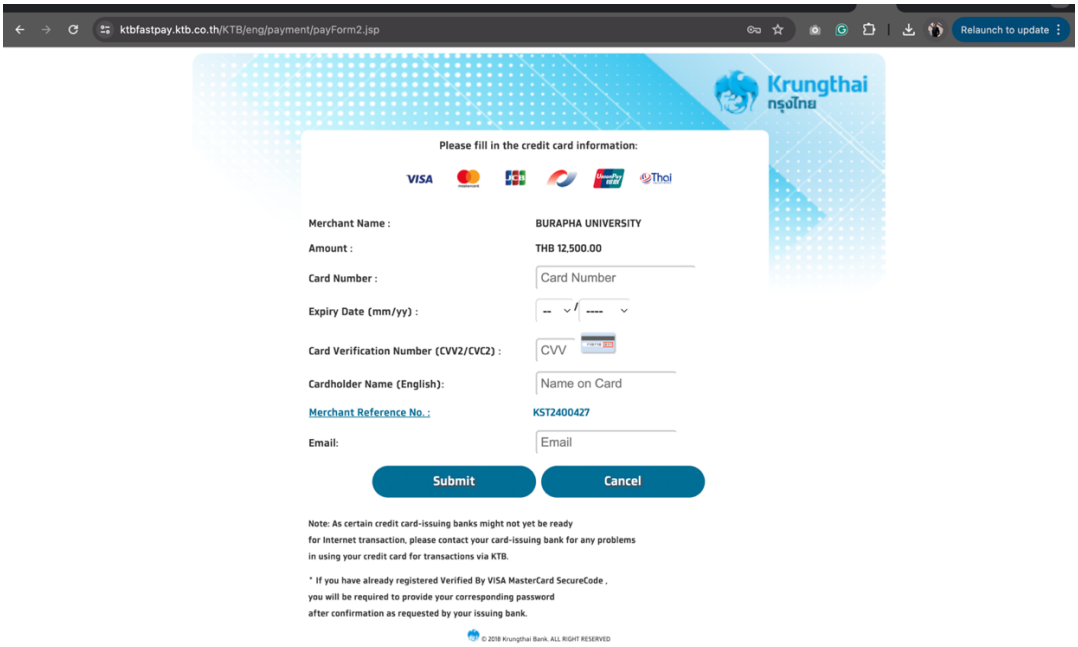
3.16 หน้าเว็บสำหรับแสดงตัวอย่างใบเสร็จรับเงิน



ภาพที่ 21 หน้าเว็บสำหรับแสดงตัวอย่างใบเสร็จรับเงิน

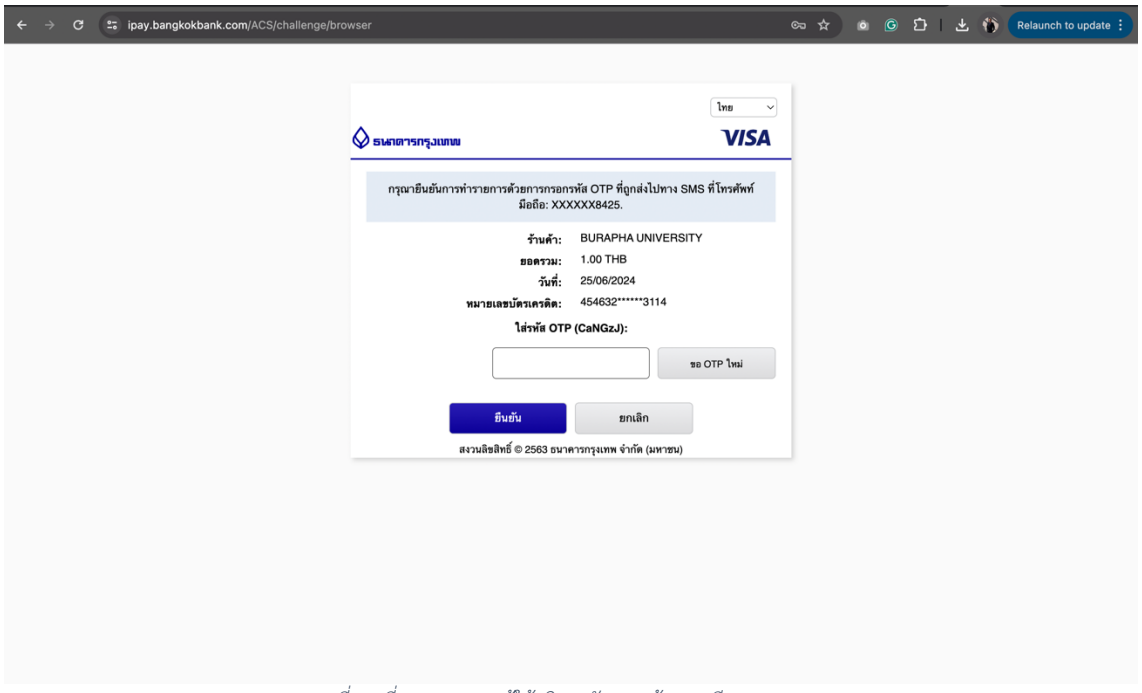
ผู้เขียนสามารถระบุรหัสลดราคา (Redeem code) เพื่อลดราคาค่าลงทะเบียนได้ (ในกรณีที่มีการตั้งค่าให้มีการใส่ redeem code ได้ในขั้นตอนการสร้างงานประชุมวิชาการ) นอกจากนี้ผู้ลงทะเบียนสามารถเลือกรูปแบบการชำระเงินได้ ซึ่งประกอบไปด้วยการชำระเงินผ่านบัตรเครดิต/เดบิค และการโอนชำระผ่านธนาคาร (Wire Transfer) ซึ่งในการชำระเงินผ่านบัตรเดรดิต ระบบใช้บริการ API (Application Program Interface) ของธนาคารกรุงไทย (KTB FastPay)

3.17 หน้าเว็บ API KTB Fast Pay



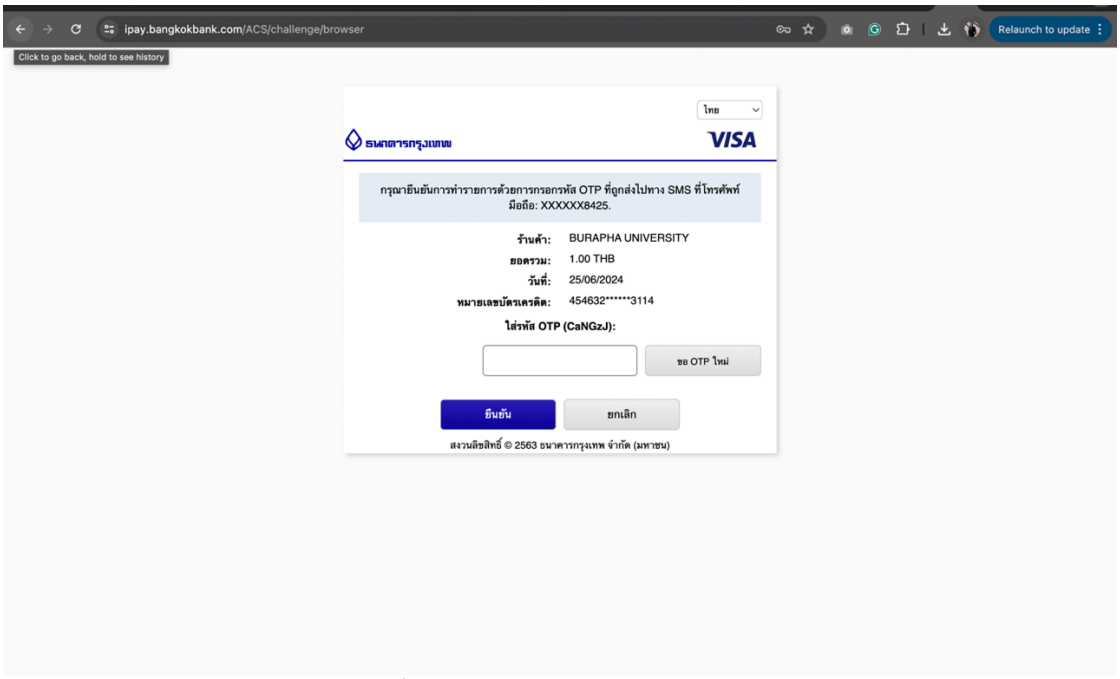
ภาพที่ 22 หน้าเว็บ API KTB Fast Pay

เมื่อระบุข้อมูลครบถ้วนแล้ว FastPay จะ Redirect ไปที่ Gateway ของผู้ให้บริการบัตรของผู้ลงทะเบียน (ในตัวอย่างเป็นหารใช้บัตรเครดิต VISA ของธนาคารกรุงเทพ)



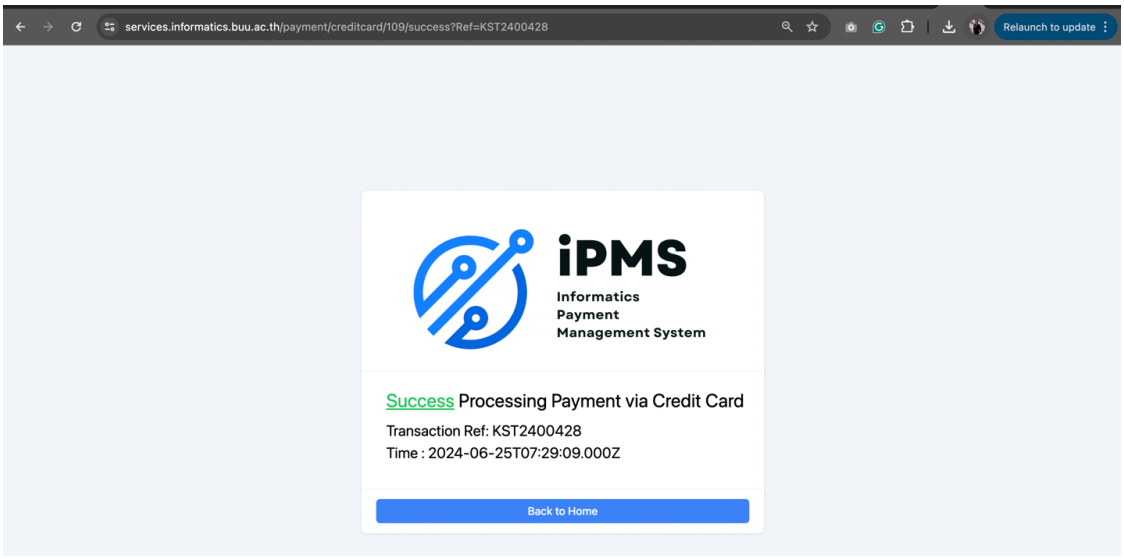
ภาพที่ 23 ที่ Gateway ของผู้ให้บริการบัตรของผู้ลงทะเบียน

ผู้ให้บริการจะส่งรหัส OTP มาที่ผู้ลงทะเบียนเพื่อเป็นการยืนยันตัวตน



ภาพที่ 24 หน้าเว็บสำหรับขอรหัส OTP ของผู้ให้บริการ

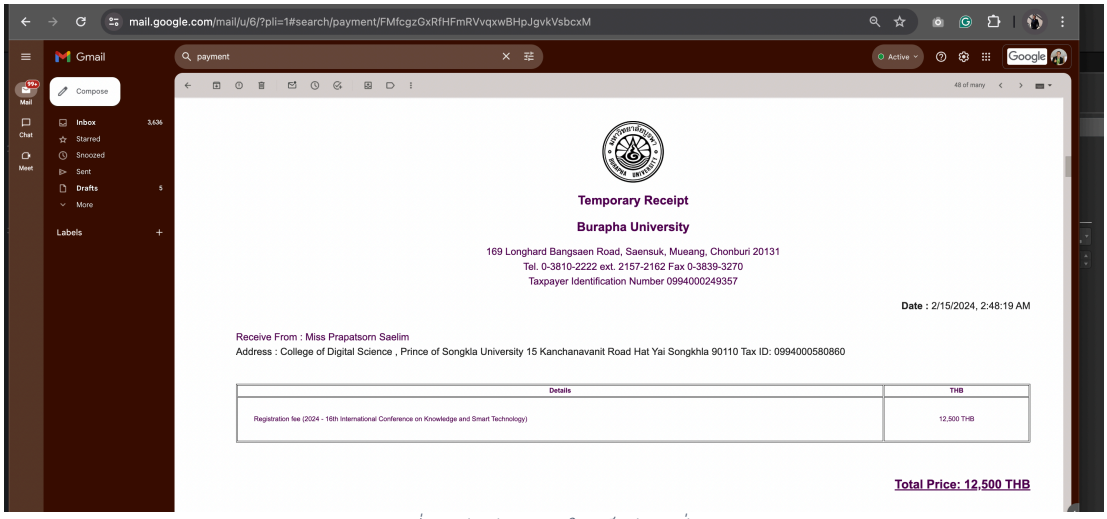
เมื่อชำระเงินผ่านบัตรเครดิตแล้วระบบจะ Redirect กลับหน้ายืนยันชำระเงิน



ภาพที่ 25 หน้าเว็บสำหรับยืนยันผลการชำระเงิน

เมื่อผู้ลงทะเบียนชำระเงินสำเร็จ ระบบ ฯ จะส่ง Email ใบเสร็จรับเงินชั่วคราว (Temporary Receipt) ให้ผู้ลงทะเบียน

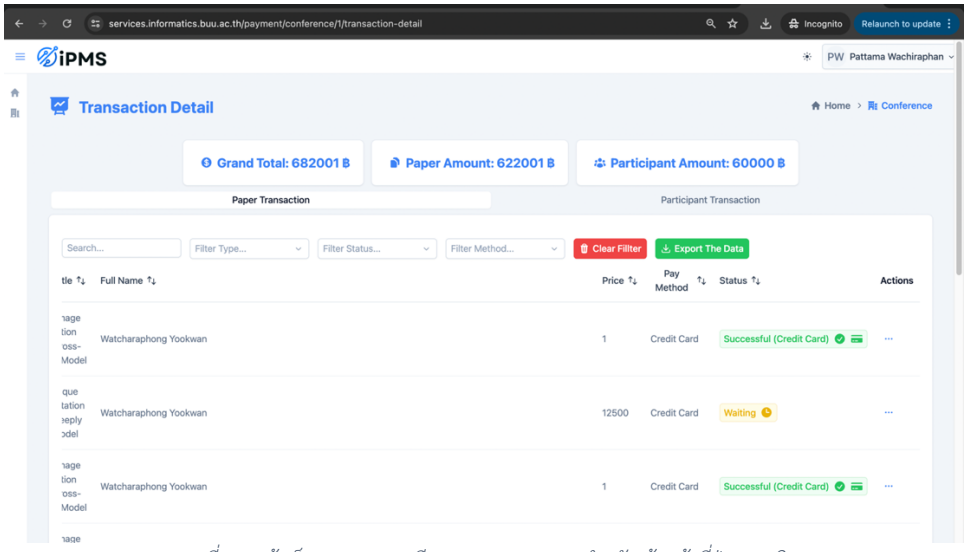
3.18 ตัวอย่าง Email ใบเสร็จรับเงินชั่วคราว



ภาพที่ 26 ตัวอย่าง Email ใบเสร็จรับเงินชั่วคราว

Email ฉบับที่ส่งออกจากระบบเป็นการยืนยันการชำระเงิน (ยังไม่ใช่ใบเสร็จรับเงินตัวขจริง) ซึ่งใบเสร็จรับเงินจะถูกออกโดยเจ้าหน้าที่การเงินในขั้นตอนการตรวจสอบยิดชำระเงินเสร็จสิ้นแล้ว

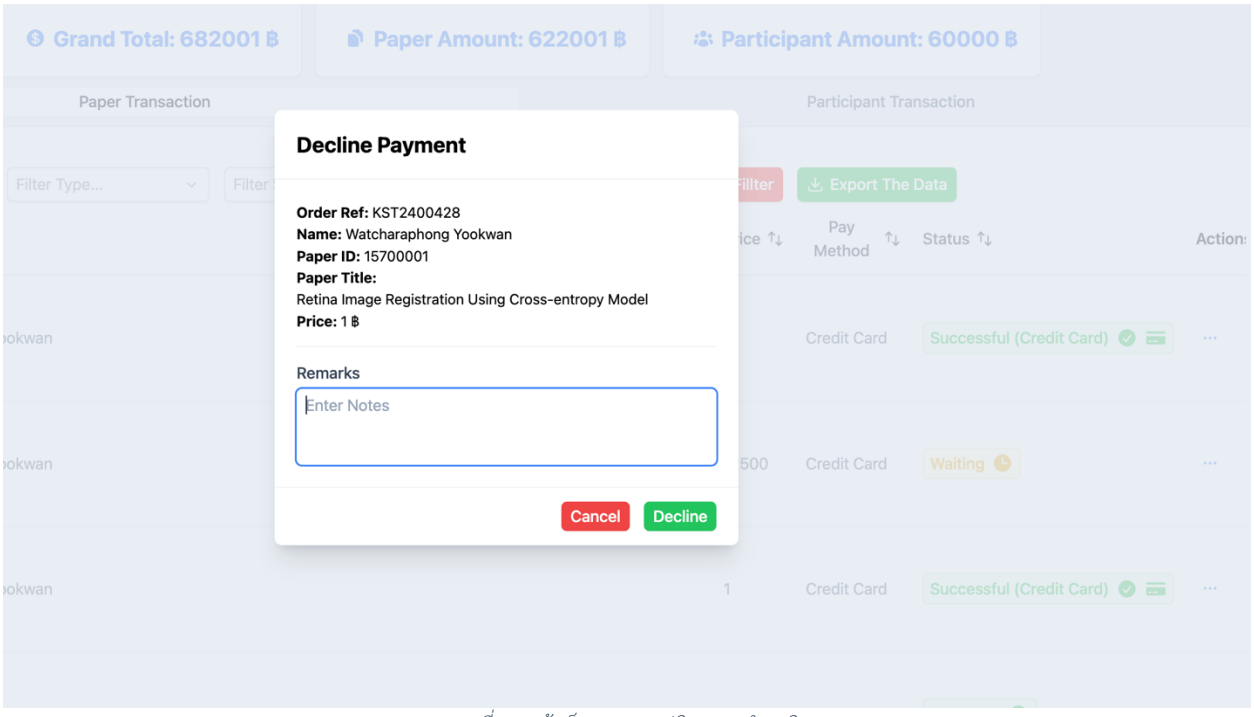
3.19 หน้าเว็บแสดงรายละเอียดของ Transactions สำหรับเจ้าหน้าที่ฝ่ายการเงิน



ภาพที่ 27 หน้าเว็บแสดงรายละเอียดของ Transactions สำหรับเจ้าหน้าที่ฝ่ายการเงิน

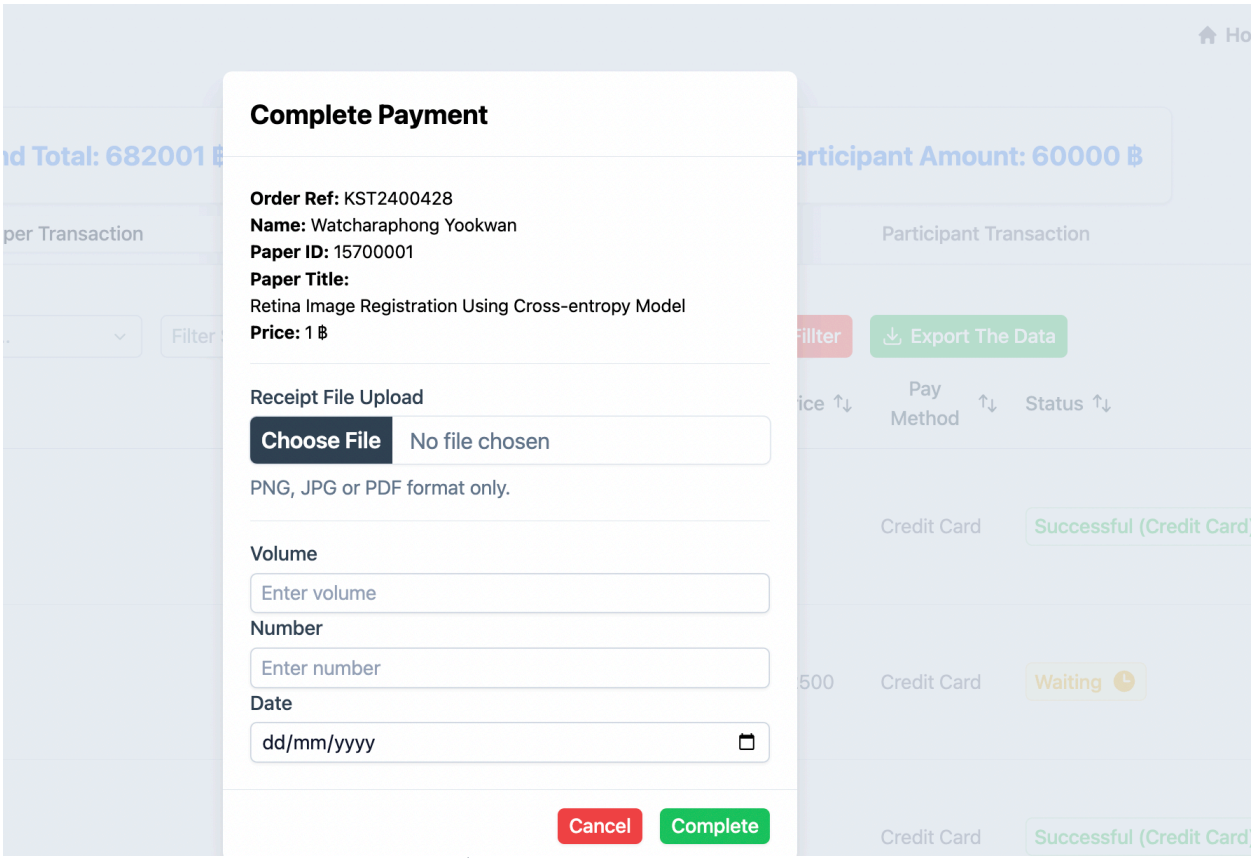
เจ้าหน้าที่ฝ่ายการเงินสามารถดูรายะเอียดของ Transactions ที่มีการชำระเงินเข้ามาได้ และสามารถเปลี่ยนสถานะของ Transaction เมื่อมีการตรวจสอบเสร็จเรียบร้อยแล้วได้ โดยสถานะจะประกอบด้วย Confirm

(ยืนยันเพื่อรออกใบเสร็จรับเงิน) Decline (ปฏิเสธการชำระเงิน) และ Complete (ออกใบเสร็จรับเงินแล้ว) เมื่อมีการเปลี่ยนสถานะ Transaction จะมี Email แจ้งผู้ลงทะเบียน ถ้าหากหลังจากเจ้าหน้าที่ฝ่ายการเงินตรวจสอบแล้วพบความผิดพลาดสามารถแจ้งผู้ลงทะเบียนกลับทาง Email ผ่านระบบได้



ภาพที่ 28 หน้าเว็บแสดงการปฏิเสธการชำระเงิน

ในกรณีที่มีการตรวจสอบการชำระเงินเรียบร้อยแล้ว เจ้าหน้าที่ฝ่ายการเงินออกใบเสร็จรับเงินตัวจริงแล้ว สามารถอัปโหลดไฟล์ใบเสร็จรับเงินตัวจริงขึ้นระบบแล้ว ระบบจะส่ง Email แจ้งผู้ทะเบียนอย่างอัตโนมัติ

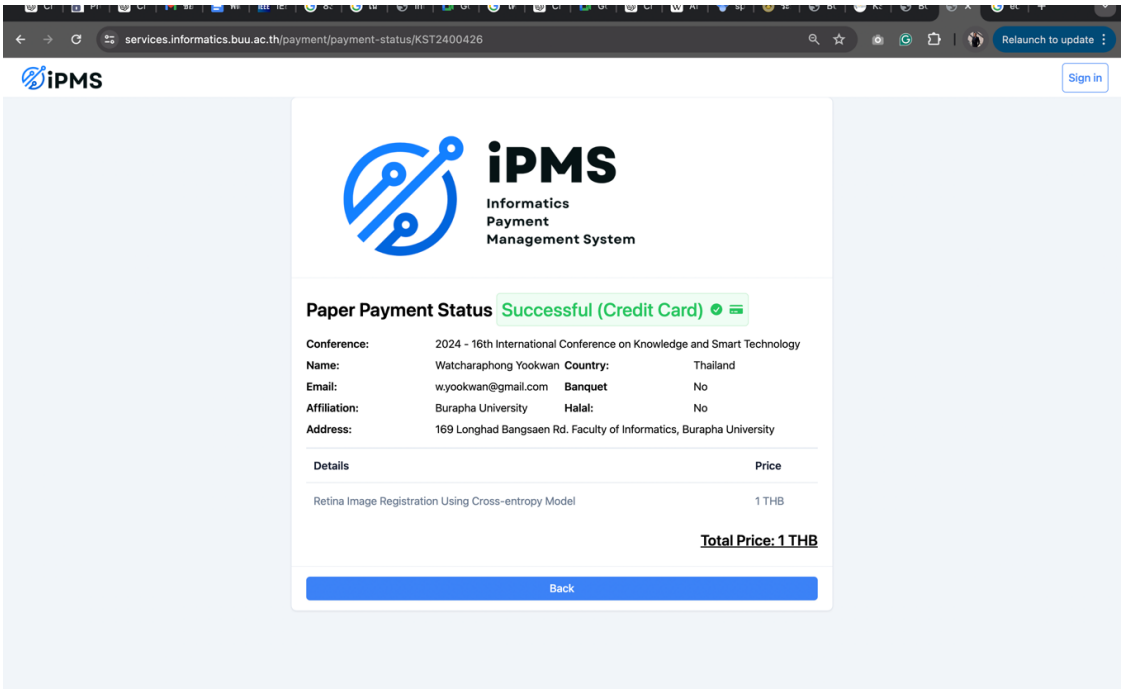


ภาพที่ 29 หน้าเว็บสำหรับยืนยันการออกใบเสร็จรับเงิน

3.20 หน้าเว็บแสดงหน้าตรวจสอบสถานะของการลงทะเบียน

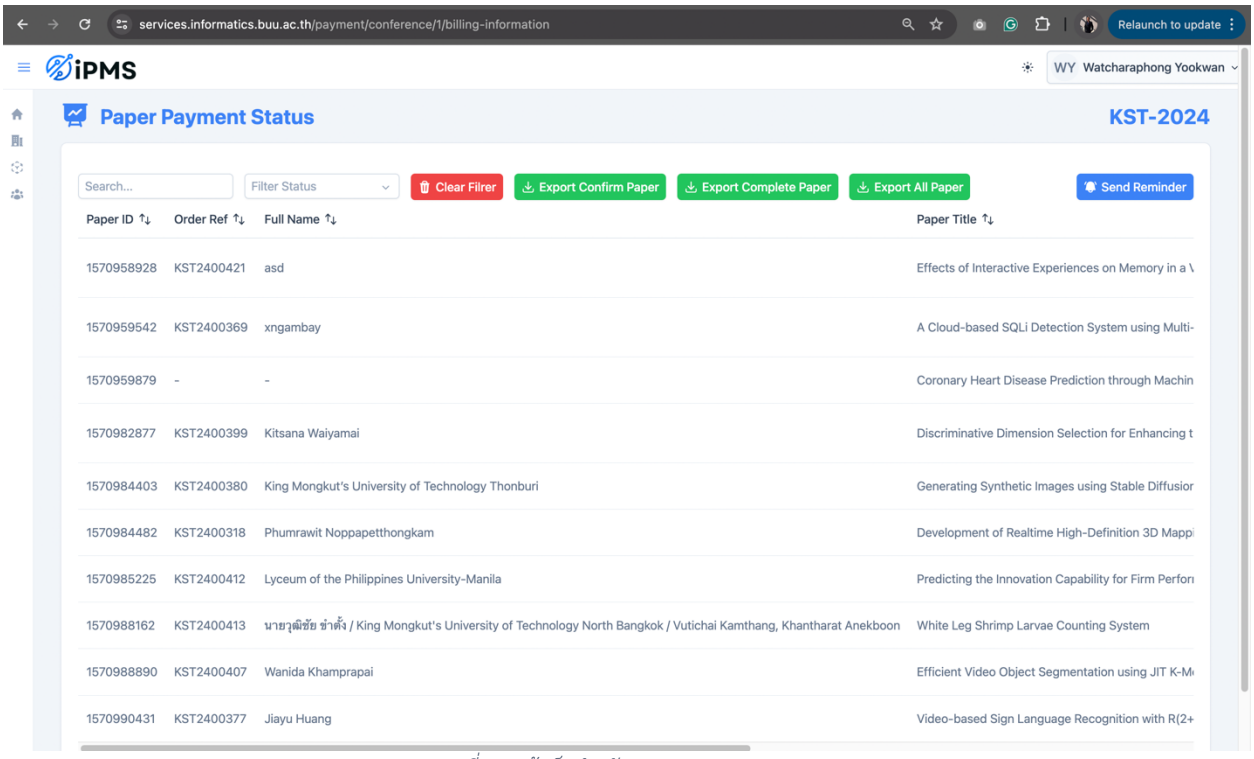
ภาพที่ 30 หน้าเว็บแสดงหน้าตรวจสอบสถานะของการลงทะเบียน

ผู้ลงทะเบียนสามารถนำหมายเลขติดตามที่ได้รับผ่านช่องทาง Email มาตรวจสอบสถานการณ์ลงทะเบียนของตนเองได้



ภาพที่ 31 หน้าเว็บสำหรับตรวจสอบสถานะการชำระเงิน

3.21 หน้าเว็บสำหรับแสดงสถานะของบทความ



ภาพที่ 32 หน้าเว็บสำหรับแสดงสถานะของบทความ

ผู้ดูแลสามารถดูรานละเอียดของสถานะของบทความที่ทำการลงทะเบียนแล้วและยังไม่ลงทะเบียนได้ พร้อมทั้งสามารถ Export ข้อมูลเป็นไฟล์ Spreadsheet เพื่อทำรายงานและใบลงนามได้ นอกจากนี้ยังสามารถส่งแจ้งเตือนสำหรับบทความที่ยังไม่ลงทะเบียนได้

3.22 ความปลอดภัยของผู้ใช้งานระบบ

ระบบชำระเงินค่าลงทะเบียนเข้าร่วมงานประชุมวิชาการที่มีความปลอดภัยเป็นสิ่งสำคัญอย่างยิ่ง เนื่องจากเกี่ยวข้องกับข้อมูลส่วนบุคคลและข้อมูลทางการเงินของผู้ใช้ ซึ่งการใช้ FastAPI สามารถช่วยให้ระบบมีความปลอดภัยและมีประสิทธิภาพได้ดังนี้

การจัดการข้อมูลผู้ใช้และการยืนยันตัวตน

- **การเข้ารหัสข้อมูล (Encryption):** FastAPI สนับสนุนการเข้ารหัสข้อมูลในการส่งผ่านระหว่างเซิร์ฟเวอร์และผู้ใช้ผ่าน HTTPS ซึ่งช่วยป้องกันการดักจับข้อมูลระหว่างการส่งข้อมูล
- **การยืนยันตัวตน (Authentication):** FastAPI สามารถใช้ JWT (JSON Web Token) หรือ OAuth2 ในการยืนยันตัวตนของผู้ใช้ ซึ่งเป็นมาตรการที่ช่วยป้องกันการเข้าถึงระบบโดยไม่ได้รับอนุญาต
- **การจัดการเซสชั่น (Session Management):** การใช้ token-based authentication สามารถช่วยจัดการเซสชั่นของผู้ใช้ได้อย่างปลอดภัย ลดความเสี่ยงจากการโจมตีแบบ session hijacking

การป้องกันการโจมตีและการเข้าถึงโดยไม่ได้รับอนุญาต

- **การป้องกันการโจมตีแบบ SQL Injection:** FastAPI สามารถใช้ ORM (Object-Relational Mapping) เช่น SQLAlchemy ที่มีการป้องกันการโจมตีแบบ SQL Injection
- **การป้องกัน Cross-Site Request Forgery (CSRF):** การใช้ token-based authentication เช่น JWT ช่วยลดความเสี่ยงจากการโจมตีแบบ CSRF
- **การควบคุมการเข้าถึง (Access Control):** การกำหนดสิทธิ์การเข้าถึงของผู้ใช้ตามบทบาท (Role-Based Access Control) ช่วยให้การเข้าถึงข้อมูลและฟังก์ชันต่างๆ ในระบบเป็นไปอย่างปลอดภัย

การป้องกันข้อมูลสูญหายและการสำรองข้อมูล

- **การสำรองข้อมูล (Data Backup):** FastAPI สามารถตั้งค่าให้มีการสำรองข้อมูลอย่างสม่ำเสมอเพื่อลดความเสี่ยงจากการสูญหายของข้อมูล
- **การเข้ารหัสข้อมูลที่จัดเก็บ (Data-at-Rest Encryption):** ข้อมูลที่จัดเก็บในฐานข้อมูลควรถูกเข้ารหัสเพื่อป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต

การรักษาความปลอดภัยของโค้ดและการปรับปรุงระบบ

- **การตรวจสอบความปลอดภัยของโค้ด (Code Security Audit):** การตรวจสอบความปลอดภัยของโค้ดอย่างสม่ำเสมอช่วยป้องกันการเกิดช่องโหว่ในระบบ
- **การปรับปรุงและแพทช์ (Updates and Patches):** การปรับปรุงระบบและการติดตั้งแพทช์เพื่อปิดช่องโหว่ที่พบอยู่เป็นสิ่งสำคัญในการรักษาความปลอดภัย

การตรวจสอบและการเฝ้าระวัง (Monitoring and Logging)

- **การบันทึกกิจกรรม (Logging):** FastAPI รองรับการบันทึกกิจกรรมต่างๆ ในระบบ ซึ่งช่วยให้สามารถตรวจสอบและติดตามเหตุการณ์ที่อาจเป็นภัยคุกคามได้
- **การเฝ้าระวังระบบ (Monitoring):** การติดตามและเฝ้าระวังระบบอย่างต่อเนื่องช่วยให้สามารถตรวจพบและตอบสนองต่อภัยคุกคามได้อย่างรวดเร็ว

บทที่ 4

ผลการดำเนินงาน

4.1 การทดสอบ User Acceptance Test (UAT)

ในการทดสอบการทำงานของระบบในขั้นสุดท้าย คณะผู้วิจัยใช้การทดสอบแบบ UAT (User Acceptance Test) เพื่อเป็นการทดสอบก่อนเริ่มใช้งานจริง โดย UAT หรือ **User Acceptance Testing** คือ กระบวนการทดสอบที่มีวัตถุประสงค์เพื่อให้ผู้ใช้งานหรือผู้ที่เกี่ยวข้องตรวจสอบและยอมรับซอฟต์แวร์ก่อนที่จะเปิดตัวใช้งานจริง UAT เป็นขั้นตอนสุดท้ายของกระบวนการทดสอบซอฟต์แวร์ โดยมีจุดประสงค์เพื่อยืนยันว่าซอฟต์แวร์ตอบสนองความต้องการและความคาดหวังของผู้ใช้งานจริง

ลักษณะสำคัญของ UAT

- ดำเนินการโดยผู้ใช้งาน (End Users): UAT มักจะดำเนินการโดยผู้ใช้งานจริงหรือตัวแทนของผู้ใช้งาน เช่น ลูกค้า, พนักงาน, หรือผู้มีส่วนเกี่ยวข้องอื่นๆ ที่จะใช้ซอฟต์แวร์ในชีวิตประจำวัน
- ทดสอบในสภาพแวดล้อมจริง (Realistic Environment): การทดสอบจะถูกดำเนินการในสภาพแวดล้อมที่คล้ายกับสภาพแวดล้อมการใช้งานจริงมากที่สุด เพื่อให้แน่ใจว่าซอฟต์แวร์สามารถทำงานได้ตามที่คาดหวังในสถานการณ์จริง
- ยืนยันความถูกต้องตามข้อกำหนด (Validation against Requirements): การทดสอบนี้จะตรวจสอบว่าซอฟต์แวร์ตอบสนองตามข้อกำหนดและความต้องการที่ได้ระบุไว้ในเอกสารข้อกำหนด (Requirements Document)
- การตรวจสอบความพร้อมใช้งาน (Readiness for Production): เป็นการตรวจสอบว่าซอฟต์แวร์พร้อมที่จะถูกนำไปใช้งานจริงหรือไม่ โดยไม่มีข้อบกพร่องสำคัญที่อาจส่งผลกระทบต่อการทำงาน

ตารางที่ 1 ผลการทดสอบ UAT

ลำดับที่	ฟังก์ชั่นการทำงาน	ประเภทการทดสอบ	ผลการทดสอบ
1	ผู้เขียนบทความสามารถระบุรายชื่อและจำนวนผู้ที่จะเข้าร่วมงานประชุมพร้อมทราบจำนวนเงินที่ต้องชำระอย่างอัตโนมัติ	Functional Test/ ทดสอบการใช้งานจริง	PASS

ลำดับที่	ฟังก์ชั่นการทำงาน	ประเภทการทดสอบ	ผลการทดสอบ
2	ผู้เขียนบทความสามารถชำระเงินผ่านบัตรเครดิตและโอนเงินผ่านธนาคารพร้อมทั้งระบุรายละเอียดในใบเสร็จรับเงินได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
3	ผู้เขียนบทความสามารถระบุรายละเอียดของผู้เข้าร่วมและนำเสนองานประชุมวิชาการได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
4	ผู้เขียนบทความสามารถอัปโหลดเอกสารใช้ประกอบการส่งบทความสำหรับตีพิมพ์ได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
5	ผู้เข้าร่วมสามารถระบุข้อมูลต่างๆ ที่ใช้ในการเข้าร่วมงานประชุมวิชาการ	Functional Test/ ทดสอบการใช้งานจริง	PASS
6	ผู้เข้าร่วมสามารถชำระเงินผ่านบัตรเครดิตและโอนเงินผ่านธนาคารได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
7	เมื่อชำระเงินแล้วผู้เข้าร่วมจะได้รับอีเมล์ยืนยันการชำระเงิน	Functional Test/ ทดสอบการใช้งานจริง	PASS
8	ผู้ดูแลระบบสามารถสร้าง ลบ แก้ไข งานประชุมวิชาการได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
9	ผู้ดูแลระบบสามารถเพิ่ม ลบ แก้ไข บทความในงานประชุมวิชาการได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
10	ผู้ดูแลระบบสามารถดูข้อมูลการลงทะเบียนได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
11	ผู้ดูแลระบบสามารถยืนยันหรือปฏิเสธข้อมูลและเอกสารการลงทะเบียนได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
12	ผู้ดูแลระบบสามารถเพิ่ม ลบ แก้ไข เจ้าหน้าที่ฝ่ายการเงินและเจ้าหน้าที่บรรณาธิการได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
13	เจ้าหน้าที่ฝ่ายการเงินสามารถตรวจสอบข้อมูลการชำระเงินของผู้เขียนบทความและผู้เข้าร่วมได้	Functional Test/ ทดสอบการใช้งานจริง	PASS
14	เจ้าหน้าที่ฝ่ายการเงินสามารถยืนยันหรือปฏิเสธการชำระเงินได้	Functional Test/ ทดสอบการใช้งานจริง	PASS

ลำดับที่	ฟังก์ชั่นการทำงาน	ประเภทการทดสอบ	ผลการทดสอบ
15	เจ้าหน้าที่ฝ่ายการเงินสามารถพิมพ์รายงานการชำระเงินได้	Functional Test/ ทดสอบการใช้งานจริง	PASS

4.2 ความพึงพอใจของผู้ใช้

การประเมินความพึงพอใจของผู้ใช้งานระบบชำระเงินค่าลงทะเบียนเข้าร่วมงานประชุมวิชาการเป็นส่วนสำคัญในการตรวจสอบความมีประสิทธิภาพและความเป็นมิตรต่อผู้ใช้งานของระบบ เพื่อรวบรวมข้อมูลที่มีประโยชน์ในการปรับปรุงระบบต่อไป ตารางด้านล่างนี้เป็นตัวอย่างของตารางประเมินความพึงพอใจของผู้ใช้งาน โดยรวบรวมข้อมูลจากผู้ใช้งานระบบจำนวน 40 คน ทั้งคนไทยและต่างชาติ

ตารางที่ 2 ผลการสำรวจความพึงพอใจ

ลำดับ	หัวข้อประเมิน	ความพึงพอใจ
1	ความสะดวกในการเข้าใช้งานระบบ	4.4
2	ความชัดเจนของข้อมูลและขั้นตอนการชำระเงิน	4.9
3	ความรวดเร็วในการทำรายการชำระเงิน	4.8
4	ความปลอดภัยของข้อมูลส่วนตัว	4.9
5	การยืนยันการชำระเงินและการแจ้งเตือน	5
6	การช่วยเหลือและการสนับสนุนจากทีมงาน	4.1
7	ความสามารถในการแก้ไขปัญหาที่พบ	4.1

จากตารางแสดงความพึงพอใจการใช้งานระบบชำระเงินค่าลงทะเบียนงานประชุมวิชาการ มีการให้คะแนนในแต่ละหัวข้อประเมินดังนี้

ความสะดวกในการเข้าใช้งานระบบ (4.4)

- คะแนนความพึงพอใจในหัวข้อนี้อยู่ในระดับสูง แต่ยังคงมีช่องว่างให้ปรับปรุง เพื่อให้ผู้ใช้งานสามารถเข้าใช้งานระบบได้อย่างสะดวกสบายยิ่งขึ้น การปรับปรุง UI/UX และการให้คำแนะนำการใช้งานที่ชัดเจนอาจช่วยให้คะแนนสูงขึ้นได้

ความชัดเจนของข้อมูลและขั้นตอนการชำระเงิน (4.9)

- คะแนนในหัวข้อนี้สูงมาก แสดงว่าผู้ใช้งานส่วนใหญ่พึงพอใจในความชัดเจนของข้อมูลและขั้นตอนการชำระเงิน ซึ่งเป็นสิ่งที่ดี เนื่องจากความชัดเจนในกระบวนการจะช่วยลดข้อผิดพลาดและปัญหาที่อาจเกิดขึ้น

ความรวดเร็วในการทำรายการชำระเงิน (4.8)

- คะแนนในหัวข้อนี้สูง แสดงให้เห็นว่าผู้ใช้งานพึงพอใจในความรวดเร็วในการทำรายการ การคงรักษาประสิทธิภาพและความรวดเร็วในระบบจะเป็นปัจจัยที่สำคัญในการรักษาความพึงพอใจของผู้ใช้งาน

ความปลอดภัยของข้อมูลส่วนตัว (4.9)

- ความพึงพอใจในหัวข้อนี้อยู่ในระดับสูง ซึ่งเป็นสิ่งที่ดีมาก เนื่องจากความปลอดภัยของข้อมูลส่วนตัวเป็นสิ่งที่สำคัญอย่างยิ่ง การรักษามาตรการความปลอดภัยและการปกป้องข้อมูลของผู้ใช้งานอย่างต่อเนื่องจะช่วยเสริมสร้างความเชื่อมั่นในระบบ

การยืนยันการชำระเงินและการแจ้งเตือน (5)

- คะแนนเต็มในหัวข้อนี้แสดงให้เห็นว่าผู้ใช้งานมีความพึงพอใจสูงสุดในการยืนยันการชำระเงินและการแจ้งเตือน ซึ่งหมายความว่าระบบการแจ้งเตือนและการยืนยันนั้นทำงานได้อย่างมีประสิทธิภาพ

การช่วยเหลือและการสนับสนุนจากทีมงาน (4.1)

- คะแนนในหัวข้อนี้ค่อนข้างต่ำกว่าหัวข้ออื่น ๆ ซึ่งแสดงว่าผู้ใช้งานอาจมีปัญหาหรือความไม่พึงพอใจในการได้รับความช่วยเหลือและการสนับสนุนจากทีมงาน การปรับปรุงการตอบสนองและการให้บริการลูกค้าอาจช่วยเพิ่มคะแนนในหัวข้อนี้ได้

ความสามารถในการแก้ไขปัญหาที่พบ (4.1)

- คะแนนในหัวข้อนี้ก็อยู่ในระดับต่ำเช่นกัน แสดงว่าผู้ใช้งานอาจพบปัญหาในการแก้ไขปัญหาที่เกิดขึ้น การพัฒนาระบบสนับสนุนและการฝึกอบรมทีมงานเพื่อให้สามารถแก้ไขปัญหาได้อย่างรวดเร็วและมีประสิทธิภาพจะเป็นประโยชน์ในการปรับปรุงคะแนนในหัวข้อนี้

จากผลการประเมินความพึงพอใจ พบว่าผู้ใช้งานมีความพึงพอใจในหลาย ๆ ด้านของระบบชำระเงินค่าลงทะเบียนเข้าร่วมงานประชุมวิชาการ โดยเฉพาะความชัดเจนของข้อมูลและขั้นตอนการชำระเงิน ความรวดเร็วในการทำรายการ และความปลอดภัยของข้อมูลส่วนตัว อย่างไรก็ตาม มีบางด้านที่ต้องการการปรับปรุง เช่น การช่วยเหลือและการสนับสนุนจากทีมงาน และความสามารถในการแก้ไขปัญหาที่พบ การปรับปรุงในด้านเหล่านี้จะช่วยเพิ่มความพึงพอใจของผู้ใช้งานได้ในอนาคต

บทที่ 5

สรุปผลการดำเนินงาน

โครงการวิจัยพัฒนาระบบชำระเงินค่าลงทะเบียนงานประชุมวิชาการเป็นความพยายามในการปรับปรุงกระบวนการจัดการการชำระเงินให้มีประสิทธิภาพมากยิ่งขึ้น ผ่านการวางแผนและการพัฒนาที่คำนึงถึงประสบการณ์และความพึงพอใจของผู้ใช้งานเป็นสำคัญ โครงการนี้เริ่มต้นจากการวิเคราะห์ปัญหาและความต้องการของผู้เข้าร่วมงานประชุมวิชาการ จากนั้นดำเนินการพัฒนาและทดสอบระบบเพื่อให้มั่นใจว่าระบบที่พัฒนาขึ้นนั้นสามารถตอบสนองความต้องการได้อย่างครบถ้วน

ในขั้นตอนการวิเคราะห์ ทีมงานได้สำรวจและรวบรวมข้อมูลจากผู้ใช้งานเพื่อทำความเข้าใจถึงปัญหาที่พวกเขาประสบในการชำระเงินค่าลงทะเบียน เช่น ความล่าช้า ความไม่สะดวกในการใช้งาน และความไม่มั่นใจในความปลอดภัยของข้อมูลส่วนตัว ข้อมูลเหล่านี้ถูกนำมาใช้ในการออกแบบระบบใหม่ที่มีความทันสมัยและตอบสนองต่อความต้องการของผู้ใช้งาน

ระบบชำระเงินค่าลงทะเบียนที่พัฒนาขึ้นใหม่มีคุณสมบัติที่สำคัญหลายประการ ได้แก่ การดำเนินการชำระเงินที่รวดเร็วและมีประสิทธิภาพ การรักษาความปลอดภัยของข้อมูลส่วนตัว การยืนยันการชำระเงินและการแจ้งเตือนที่ชัดเจน นอกจากนี้ ระบบยังถูกออกแบบให้ใช้งานง่าย ไม่ซับซ้อน เพื่อให้ผู้ใช้งานสามารถดำเนินการได้ด้วยตนเองอย่างสะดวกสบาย

หลังจากการพัฒนาระบบเสร็จสิ้น ทีมงานได้ดำเนินการทดสอบระบบกับกลุ่มผู้ใช้งานจริงและเก็บรวบรวมข้อมูลความพึงพอใจ ผลการประเมินความพึงพอใจจากผู้ใช้งานพบว่าระบบที่พัฒนาขึ้นนั้นมีความพึงพอใจในระดับสูง แสดงให้เห็นว่าระบบสามารถตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างดี ทั้งในด้านความสะดวก ความรวดเร็ว และความปลอดภัย

อย่างไรก็ตาม ยังมีบางด้านที่ต้องการการปรับปรุง เช่น การช่วยเหลือและการสนับสนุนจากทีมงาน รวมถึงความสามารถในการแก้ไขปัญหาที่พบ จากการประเมินพบว่าผู้ใช้งานยังต้องการการช่วยเหลือจากทีมงานเพิ่มเติมในกรณีที่เกิดปัญหาในการใช้งาน การปรับปรุงในด้านนี้จะช่วยเสริมสร้างความพึงพอใจของผู้ใช้งานได้ในอนาคต

การเก็บรวบรวมและวิเคราะห์ข้อมูลจากการประเมินความพึงพอใจนี้จะเป็นประโยชน์อย่างยิ่งในการปรับปรุงและพัฒนาระบบให้มีความสมบูรณ์และตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างเต็มที่ในอนาคต ทำให้การชำระเงินค่าลงทะเบียนงานประชุมวิชาการเป็นไปอย่างราบรื่นและมีประสิทธิภาพสูงสุด การพัฒนาระบบในครั้งนี้ถือเป็นก้าวสำคัญในการเพิ่มประสิทธิภาพและความพึงพอใจในการใช้งานระบบชำระเงินค่าลงทะเบียนงานประชุมวิชาการ อีกทั้งยังช่วยส่งเสริมภาพลักษณ์ที่ดีของงานประชุมวิชาการในอนาคต

บรรณานุกรม

Programming language. (2024, June 23). In *Wikipedia*. https://en.wikipedia.org/wiki/Programming_language

Voron, F. (2023). *Building Data Science Applications with FastAPI: Develop, manage, and deploy efficient machine learning applications with Python*. Packt Publishing Ltd.

Myers, J., & Copeland, R. (2015). *Essential SQLAlchemy: Mapping Python to Databases*. " O'Reilly Media, Inc.".

McGill, T. J., & Volet, S. E. (1997). A conceptual framework for analyzing students' knowledge of programming. *Journal of research on Computing in Education*, *29*(3), 276-297