การพัฒนาระบบจองใบอนุญาตการประชุมออนไลน์แบบอัจฉริยะ : นวัตกรรมเทคโนโลยีสารสนเทศเพื่อประหยัดงบประมาณและเพิ่มประสิทธิภาพองค์กร

Development of Smart Online Meeting License Booking System : IT Innovation for Cost Reduction and Enhanced Organizational Performance

นโรดม กิตติเดชานุภาพ[1*] อาทิตย์ ดีอราโซ[2] และ ภูริทรัพย์ เดชพิพัฒน์ประชา[3]

Narodom Kittidachanuphap[1*], Arthit Dueraso[2], and Bhurisub Dejpipatpracha[3]

คณะวิทยาศาสตร์เทคโนโลยีและการเกษตร มหาวิทยาลัยราชภัฏยะลา[1]

Faculty of Science, Technology and Agriculture, Yala Rajabhat University[1]

สำนักวิทยบริการและเทคโนโลยีสารสนเทศ มหาวิทยาลัยราชภัฏยะลา[2]

Office of Academic Services and Information Technology, Yala Rajabhat University[2]

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏภูเก็ต[3]

Faculty of Science and Technology, Phuket Rajabhat University[3]

Email: narodom.k@yru.ac.th

Received : July 31, 2025
Revised : October 6, 2025
Accepted : December 1, 2025

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อ 1) พัฒนาระบบจองใบอนุญาตการประชุมออนไลน์แบบรวมศูนย์ที่มีประสิทธิภาพและบุคลากรภายในมหาวิทยาลัยสามารถใช้งานได้อย่างเท่าเทียม 2) เพิ่มประสิทธิภาพการใช้งานใบอนุญาตการประชุมออนไลน์ให้เกิดประโยชน์สูงสุด และ 3) ลดต้นทุนการจัดซื้อใบอนุญาตการประชุมออนไลน์ของมหาวิทยาลัย

การดำเนินการวิจัยใช้ประชากรและกลุ่มตัวอย่างจากบุคลากรมหาวิทยาลัยราชภัฏยะลา จำนวน 832 คน แบ่งเป็น 2 กลุ่ม ได้แก่ กลุ่มตัวอย่างสำหรับการพัฒนาระบบ คัดเลือกแบบเจาะจง จำนวน 50 ราย ประกอบด้วย อาจารย์ 25 ราย เจ้าหน้าที่สายสนับสนุน 15 ราย และบุคลากรสายบริหาร 10 ราย และกลุ่มตัวอย่างสำหรับการประเมินความพึงพอใจ จำนวน 256 ราย ประกอบด้วย อาจารย์ เจ้าหน้าที่ และนักศึกษาระดับบัณฑิตศึกษา เครื่องมือในการวิจัย ได้แก่ ระบบที่พัฒนาด้วยวงจรการพัฒนาซอฟต์แวร์ (Software Development Life Cycle : SDLC) โดยใช้สถาปัตยกรรมเรสต์ฟูลเอพีไอ (RESTful API) ประกอบด้วยแบ็กเอนด์ที่พัฒนาด้วยลาราเวล (Laravel) และฟรอนต์เอนด์ที่พัฒนาด้วยนักซ์ (Nuxt) เชื่อมต่อกับส่วนต่อประสานโปรแกรมประยุกต์ของแพลตฟอร์ม การประชุมออนไลน์สำหรับการจัดการ

ใบอนุญาตอัตโนมัติ แบบสอบถามความพึงพอใจ 5 ระดับ และเครื่องมือทดสอบประสิทธิภาพ อพาเช่ เจเมเตอร์ (Apache JMeter) สถิติที่ใช้ในการวิจัย ได้แก่ สถิติเชิงพรรณนา (Descriptive Statistics) การวิเคราะห์เนื้อหา (Content Analysis) และการวิเคราะห์ข้อมูล แบบผสมผสาน (Mixed-Method Analysis)

ผลการวิจัยพบว่า 1) ระบบที่พัฒนาขึ้นสามารถจัดการใบอนุญาตแบบรวมศูนย์ได้อย่างมีประสิทธิภาพ โดยลดเวลาการจองจาก 30 นาทีเหลือ 3 นาที (ลดลงร้อยละ 90) และลดขั้นตอนการจองจาก 10 ขั้นตอนเหลือ 4 ขั้นตอน ระบบรองรับผู้ใช้งานพร้อมกัน 500 ราย ด้วยเวลาตอบสนองเฉลี่ย 1.2 วินาที มีความพร้อมใช้งานร้อยละ 99.8 และผู้ใช้งานสามารถเข้าถึงระบบได้อย่างเท่าเทียมผ่านระบบลงชื่อเข้าใช้เดียว (Single Sign-On : SSO) 2) ระบบเพิ่มประสิทธิภาพการใช้งานใบอนุญาตได้ร้อยละ 85 จากการจัดสรรแบบแบ่งปันและการตรวจสอบการใช้งานแบบอัตโนมัติ สามารถจัดการคิวการจอง ติดตามการใช้งาน และสร้างรายงานเชิงสถิติแบบเรียลไทม์ และ 3) ระบบลดต้นทุนการจัดซื้อใบอนุญาตลงได้ร้อยละ 60 จากการใช้ทรัพยากรอย่างมีประสิทธิภาพและการลดการซื้อใบอนุญาตที่ไม่จำเป็น การประเมินความพึงพอใจจากกลุ่มตัวอย่างผู้ใช้งาน 256 ราย ได้คะแนนเฉลี่ย 4.20 จากคะแนนเต็ม 5.00 (ร้อยละ 84) ในระดับดีมาก ผลการวิจัยนี้สามารถนำไปประยุกต์ใช้ในสถาบันการศึกษาอื่นและองค์กรที่มีความต้องการจัดการทรัพยากรเทคโนโลยีสารสนเทศแบบแบ่งปัน

คำสำคัญ: ระบบจองออนไลน์ การจัดการใบอนุญาต เว็บเซอร์วิสเรสต์ฟูล การประชุมออนไลน์

ABSTRACT

This research aims to: 1) develop an efficient centralized online meeting license booking system that enables equitable access for university personnel; 2) maximize the utilization efficiency of online meeting licenses; and 3) reduce the procurement costs of online meeting licenses for the university.

The research was conducted using a population of 832 personnel from Yala Rajabhat University. The study employed two sample groups: a purposive sample of 50 personnel for system development (comprising 25 faculty members, 15 support staff, and 10 administrative personnel), and a sample of 256 users for satisfaction evaluation (comprising faculty members, staff, and graduate students). Research instruments included a system developed using the Software Development Life Cycle (SDLC) with RESTful API architecture, consisting of a Laravel-based backend and Nuxt-based frontend. This system was integrated with the application

programming interface of the online meeting platform for automated license management. Other instruments included a 5-level satisfaction questionnaire and Apache JMeter for performance testing. Statistical analyses employed included descriptive statistics, content analysis, and mixed-method analysis.

Research findings revealed that 1) the developed system efficiently managed centralized license allocation by reducing booking time from 30 minutes to 3 minutes (a 90% reduction) and decreasing booking steps from 10 to 4 steps. The system supported 500 concurrent users with an average response time of 1.2 seconds, maintained 99.8% uptime, and provided equitable access through Single Sign-On (SSO). 2) the system increased license utilization efficiency by 85% through shared allocation and automated usage monitoring, enabling queue management, usage tracking, and real-time statistical reporting. And 3) the system reduced license procurement costs by 60% through efficient resource utilization and elimination of unnecessary license purchases. User satisfaction evaluated from the sample of 256 users averaged 4.20 out of 5.00 (84%), indicating an excellent level of satisfaction. The research findings can be applied to other educational institutions and organizations with shared information technology resource management needs.

Keywords: Online booking system, License management, RESTful Web Services, Online meetings

บทนำ

สถานการณ์การแพร่ระบาดของโรคโควิด-19 (COVID-19 Pandemic) ได้ส่งผลกระทบอย่างรุนแรงและเปลี่ยนแปลงรูปแบบการทำงาน (Work Pattern) และการจัดการศึกษา (Educational Management) ไปสู่ระบบออนไลน์ (Online System) อย่างถาวร ทำให้ความต้องการใช้งานแพลตฟอร์มการประชุมออนไลน์ (Video Conferencing Platforms) เช่น Zoom, Microsoft Teams และ Google Meet เพิ่มขึ้นอย่างก้าวกระโดด โดยเฉพาะอย่างยิ่งในสถาบันการศึกษา (Educational Institutions) ที่ต้องปรับตัวเพื่อรองรับการเรียนการสอนแบบไฮบริด (Hybrid Learning) การประชุมทางวิชาการ (Academic Meetings) และกิจกรรมบริหารจัดการต่าง ๆ (Administrative Activities) (Wang et al., 2021; Martinez & Johnson, 2020) การเปลี่ยนแปลงครั้งนี้ไม่เพียงแต่เป็นการปรับตัวชั่วคราว (Temporary Adaptation) แต่ได้กลายเป็นรูปแบบการทำงานใหม่ (New Normal) ที่คาดว่า

จะคงอยู่ต่อไปในอนาคต สะท้อนให้เห็นถึงความสำคัญของการลงทุนด้านเทคโนโลยีสารสนเทศ (Information Technology Investment) และการบริหารจัดการทรัพยากรดิจิทัล (Digital Resource Management) อย่างมีประสิทธิภาพ

การศึกษาเบื้องต้น (Preliminary Study) ของ Chen และคณะ (2020) พบว่า สถาบันการศึกษาขนาดกลางถึงขนาดใหญ่ (Medium to Large Educational Institutions) มีความต้องการใบอนุญาตการประชุมออนไลน์ (Online Meeting Licenses) เพิ่มขึ้นถึง 300% ในช่วงปี พ.ศ. 2563-2564 เมื่อเทียบกับช่วงก่อนการระบาด มหาวิทยาลัยราชภัฏยะลา (Yala Rajabhat University) ซึ่งมีบุคลากรทั้งสิ้น 832 คน ประกอบด้วยอาจารย์ เจ้าหน้าที่สายสนับสนุน และบุคลากรสายบริหาร ประสบปัญหาข้อจำกัดของใบอนุญาตการประชุมออนไลน์ที่ไม่เพียงพอต่อความต้องการใช้งานจริงจากการสำรวจเบื้องต้นระหว่างเดือนมกราคม-มีนาคม พ.ศ. 2565 พบว่า มหาวิทยาลัยมีใบอนุญาตการประชุมออนไลน์เพียง 30 บัญชี (Accounts/Licenses) ซึ่งจัดซื้อตั้งแต่ปีงบประมาณ พ.ศ. 2564 ถึง พ.ศ. 2565 แต่มีความต้องการใช้งานเกิน 200 บัญชีในช่วงเวลาเดียวกัน โดยข้อมูลจากการสำรวจการใช้งานในเดือนกุมภาพันธ์ พ.ศ. 2565 พบว่า โดยเฉพาะในช่วงเวลาทำการปกติ (08:30-16:30 น.) ซึ่งเป็นช่วงที่มีการจัดการเรียนการสอนและการประชุมมากที่สุด ปัญหานี้ก่อให้เกิดความขัดแย้งในการจองใช้งานที่ไม่เป็นระบบ การใช้งานทรัพยากรที่ไม่มีประสิทธิภาพ และภาระต้นทุนการจัดซื้อใบอนุญาตเพิ่มเติมที่สูง

เพื่อแก้ไขปัญหาดังกล่าวอย่างเป็นระบบและยั่งยืน มหาวิทยาลัยราชภัฏยะลา จึงได้ริเริ่มพัฒนาระบบจองใบอนุญาตการประชุมออนไลน์แบบรวมศูนย์ (Development of a Centralized Booking System for Online Meeting Licenses) เพื่อบริหารจัดการการใช้งานใบอนุญาต (License Management) ให้เกิดประสิทธิภาพสูงสุดและเป็นธรรม โดยประยุกต์ใช้หลักการการรวบรวมทรัพยากรร่วม และการจัดสรรแบบยืดหยุ่นในการจัดการทรัพยากรแบบเรียลไทม์ (Real-time Resource Management) ระบบนี้ได้รับการออกแบบให้สามารถจัดการคิวการจอง ติดตามการใช้งาน และสร้างรายงานเชิงสถิติ เพื่อการวางแผนและการตัดสินใจในอนาคต นอกจากนี้ยังมุ่งเน้นการสร้างความโปร่งใสในการเข้าถึงทรัพยากร ลดความขัดแย้ง และเพิ่มความพึงพอใจของผู้ใช้งาน ซึ่งจะเป็นแนวทางที่สามารถขยายผล และนำไปประยุกต์ใช้ในสถาบันการศึกษาอื่น ๆ ที่ประสบปัญหาคล้ายคลึงกันได้

วัตถุประสงค์การวิจัย

1. เพื่อพัฒนาระบบจองใบอนุญาตการประชุมออนไลน์ แบบรวมศูนย์ที่มีประสิทธิภาพ และบุคลากรภายในมหาวิทยาลัยสามารถใช้งานได้อย่างเท่าเทียม
2. เพื่อเพิ่มประสิทธิภาพการใช้งานใบอนุญาตการประชุมออนไลน์ ให้เกิดประโยชน์สูงสุด
3. เพื่อลดต้นทุนการจัดซื้อใบอนุญาตการประชุมออนไลน์ ของมหาวิทยาลัย

เอกสารงานวิจัยที่เกี่ยวข้อง

แนวคิดการจัดการทรัพยากรแบบรวมศูนย์

การจัดการทรัพยากรแบบรวมศูนย์ (Centralized Resource Management) เป็นแนวคิดที่นำมาใช้ในการบริหารจัดการทรัพยากรเทคโนโลยีสารสนเทศ (IT Resources) ให้เกิดประสิทธิภาพสูงสุด โดยเฉพาะอย่างยิ่งสำหรับทรัพยากรที่มีข้อจำกัดและต้นทุนสูง (Kumar et al., 2019) การศึกษาของ Thompson และ Johnson (2020) พบว่า การจัดการทรัพยากรแบบรวมศูนย์สามารถลดต้นทุนได้ร้อยละ 40-70 และเพิ่มประสิทธิภาพการใช้งานได้ร้อยละ 50-85

สถาปัตยกรรมเรสต์ฟูลเอพีไอ

เรสต์ฟูลเอพีไอ (RESTful API - Representational State Transfer) เป็นรูปแบบการออกแบบเว็บเซอร์วิส (Web Services) ที่ใช้หลักการของโปรโตคอลเอชทีทีพี (HTTP Protocol) ในการสื่อสารระหว่างระบบ ซึ่งมีความยืดหยุ่นสูงและสามารถขยายตัวได้ดี (Fielding, 2000) การศึกษาของ Rodriguez และ Martinez (2021) แสดงให้เห็นว่า เรสต์ฟูลเอพีไอ (RESTful API) มีประสิทธิภาพในการจัดการระบบขนาดใหญ่ได้ดีกว่าเอพีไอโซป (SOAP API) ถึงร้อยละ 40

การพัฒนาระบบด้วยกรอบงานลาราเวล

ลาราเวล (Laravel) เป็นกรอบงานพีเอชพี (PHP Framework) ที่ใช้หลักการเอ็มวีซี (MVC - Model-View-Controller) และมีความสามารถในการพัฒนาเอพีไอ (API) ที่มีประสิทธิภาพรองรับการทำการยืนยันตัวตน (Authentication) และการอนุญาต (Authorization) ได้ดี (Otwell, 2021) การศึกษาของ Anderson และคณะ (2020) พบว่า ลาราเวล (Laravel) มีประสิทธิภาพในการพัฒนาระบบจัดการทรัพยากรดีกว่ากรอบงาน (Framework) อื่น ๆ ในด้านความเร็วในการพัฒนาและการบำรุงรักษา

การพัฒนาฟรอนต์เอนด์ด้วยนักซ์

นักซ์ (Nuxt) เป็นกรอบงาน (Framework) ที่พัฒนาจากวิว (Vue) เพื่อการสร้างแอปพลิเคชันสากล (Universal Application) ที่รองรับทั้งการแสดงผลฝั่งเซิร์ฟเวอร์ (Server-Side Rendering - SSR) และการแสดงผลฝั่งไคลเอนต์ (Client-Side Rendering - CSR) (Chopin et al., 2020) การวิจัยของ Lee และ Kim (2021) แสดงให้เห็นว่า นักซ์ (Nuxt) มีประสิทธิภาพดีกว่ารีแอ็กต์ (React) ในการพัฒนาระบบที่ต้องการการเพิ่มประสิทธิภาพเครื่องมือค้นหา (SEO) และการโหลดเร็ว

วิธีการดำเนินการวิจัย

ประชากรกลุ่มตัวอย่าง

ใช้วิธีการสุ่มตัวอย่างแบบเจาะจง ประกอบด้วยอาจารย์ เจ้าหน้าที่สายสนับสนุน และบุคลากรสายบริหาร โดยคัดเลือกผู้ใช้งานจำนวน 50 ราย ตามเกณฑ์การคัดเลือกที่กำหนด ได้แก่ บุคลากรที่มี

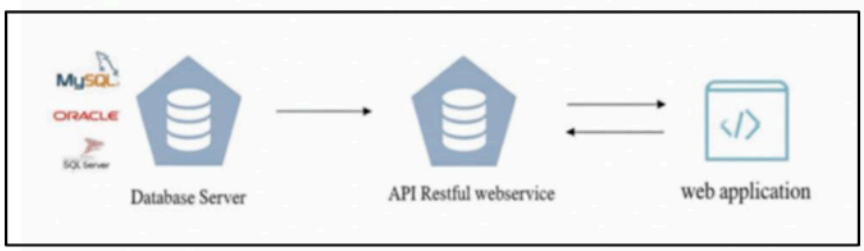
ประสบการณ์การใช้งานแพลตฟอร์มการประชุมออนไลน์อย่างสม่ำเสมอ มีความต้องการใช้งานใบอนุญาตการประชุมออนไลน์ เป็นประจำในการปฏิบัติงาน และเป็นตัวแทนจากทุกหน่วยงานภายในมหาวิทยาลัย เพื่อให้ได้ข้อมูลที่ครอบคลุมและแม่นยำจากทุกกลุ่มผู้ใช้งาน โดยใช้วิธีการสัมภาษณ์เชิงลึก

เครื่องการมือวิจัย

ขั้นตอนที่ 1 การออกแบบระบบ (System Design)

การออกแบบได้เริ่มต้นการสร้างสถาปัตยกรรมระบบ โดยใช้หลักการสถาปัตยกรรมไมโครเซอร์วิส (Microservices Architecture) เพื่อให้ระบบมีความยืดหยุ่นและสามารถขยายได้ง่าย ระบบถูกแบ่งออกเป็น 2 ส่วนหลัก คือ แบ็กเอนด์ (Backend) และ ฟรอนต์เอนด์ (Frontend) ซึ่งสื่อสารกันผ่าน เรสต์ฟูลเอพีไอ (RESTful API) การออกแบบยึดหลักการแยกส่วนความรับผิดชอบ (Separation of Concerns) เพื่อให้ระบบมีโครงสร้างที่ชัดเจนและง่ายต่อการบำรุงรักษา นอกจากนี้ยังได้ออกแบบโครงสร้างฐานข้อมูลที่มีประสิทธิภาพและจุดเชื่อมต่อเอพีไอ (API Endpoints) ที่จำเป็นสำหรับการสื่อสารระหว่างส่วนต่าง ๆ ของระบบ ผลลัพธ์จากขั้นตอนนี้ได้ถูกบันทึกไว้ในเอกสารการออกแบบระบบที่ครบถ้วน

ขั้นตอนที่ 2 การพัฒนาระบบแบ็กเอนด์ (Backend System Development)



ภาพประกอบ 1 ภาพรวมของระบบ

เมื่อการออกแบบเสร็จสิ้น ได้เริ่มต้นการพัฒนาระบบแบ็กเอนด์ (Backend) โดยใช้กรอบงานลาราเวล (Laravel Framework) ซึ่งมีโครงสร้างหลักประกอบด้วยตัวควบคุม (Controllers) สำหรับจัดการคำขอเอชทีทีพี (HTTP Requests), โมเดล (Models) สำหรับจัดการข้อมูลในฐานข้อมูล, และบริการ (Services) สำหรับเชื่อมต่อกับเอพีไอซูม (Zoom API) ระบบใช้ฐานข้อมูลมายเอสคิวแอล (MySQL Database) เป็นที่จัดเก็บข้อมูลและรองรับการยืนยันตัวตน (Authentication) ผ่านระบบลงชื่อเข้าใช้เดียว (Passport SSO) ของมหาวิทยาลัยเพื่อให้ผู้ใช้สามารถเข้าใช้งานได้อย่างสะดวกและปลอดภัย

ขั้นตอนที่ 3 การพัฒนาระบบฟรอนต์เอนด์ (Frontend System Development)

ขณะเดียวกันกับการพัฒนาแบ็กเอนด์ ได้ดำเนินการพัฒนาระบบฟรอนต์เอนด์ (Frontend) ด้วยกรอบงานนักซ์ (Nuxt Framework) ซึ่งเป็นกรอบงานวิว (Vue Framework) ที่มีความสามารถในการทำการแสดงผลฝั่งเซิร์ฟเวอร์ (Server-Side Rendering) เพื่อเพิ่มประสิทธิภาพในการโหลดหน้าเว็บ

ระบบใช้วิวติฟาย (Vuetify) เป็นไลบรารีส่วนประกอบส่วนติดต่อผู้ใช้ (UI Component Library) เพื่อให้ได้ส่วนติดต่อผู้ใช้ (User Interface) ที่สวยงามและใช้งานง่าย พร้อมรองรับการออกแบบตอบสนอง (Responsive Design) สำหรับการใช้งานบนอุปกรณ์ต่าง ๆ ทั้งคอมพิวเตอร์ แท็บเล็ต และโทรศัพท์มือถือ

ขั้นตอนที่ 4 การพัฒนาฟีเจอร์หลัก (Core Features Development)

หลังจากที่โครงสร้างพื้นฐานของระบบเสร็จสิ้น ได้เริ่มพัฒนาฟีเจอร์หลักของระบบ โดยเริ่มต้นด้วยระบบจัดการใบอนุญาตซูม (Zoom License) ที่สามารถทำงานได้อย่างอัตโนมัติผ่านการเรียกใช้เอพีไอซูม (Zoom API) ประกอบด้วยการเพิ่มและลบใบอนุญาต (License Management) การปรับขนาดใบอนุญาต (License Scaling) และการตรวจสอบสถานะใบอนุญาต (License Status Monitoring) ตามด้วยการพัฒนาระบบจองห้องประชุมที่มีความยืดหยุ่นสูง ซึ่งประกอบด้วยการสร้างห้องประชุมตามเวลาที่กำหนด การจองใบอนุญาตล่วงหน้า การยกเลิกการจองและคืนใบอนุญาต และการแจ้งเตือนอัตโนมัติ (Automatic Notification) สุดท้ายคือการพัฒนาระบบบริหารจัดการผู้ใช้งานที่ครอบคลุมการสร้าง และจัดการบัญชีผู้ใช้งาน การจัดสิทธิ์การเข้าถึงระบบตามบทบาท, และการติดตามการใช้งานเพื่อวิเคราะห์ประสิทธิภาพ โดยตัวอย่างส่วนต่อประสานโปรแกรมประยุกต์ (Application Program Interface: API) เชื่อมต่อกับ zoom.us โดยมีตัวอย่างคำสั่ง ดังภาพประกอบ 2 - 5

คำสั่งเรียกดูข้อมูลห้องประชุมทั้งหมด ภาพแสดงภาพแสดงฟังก์ชัน getMeetings() ที่ใช้ HTTP GET Request ผ่าน Zoom API endpoint /users/{userId}/meetings เพื่อดึงรายการห้องประชุมทั้งหมดของผู้ใช้ โดยส่ง Authorization Token ผ่าน Header และได้ผลลัพธ์กลับมาเป็น JSON format ที่มีข้อมูลเกี่ยวกับการประชุมแต่ละห้อง เช่น Meeting ID, หัวข้อ, เวลาเริ่มต้นและลิงก์เข้าร่วมประชุม

```php
public function getMeetings()
{
    return $this->_http->get($this->zoomUrl . 'users/' . Auth::user()->passport->email . '/meetings')->json();
}
```

ภาพประกอบ 2 คำสั่งเรียกดูข้อมูลห้องประชุมทั้งหมด

คำสั่งเรียกดูข้อมูลห้องประชุมทั้งหมดตามวันและเวลา ภาพแสดงฟังก์ชัน getMeetingsByDateTime($startDate, $endDate) ที่เพิ่ม Query Parameters (type, from, to, page_size) เข้าไปใน GET Request เพื่อกรองห้องประชุมตามช่วงวันเวลาที่ต้องการ จากนั้นใช้ฟังก์ชัน filterMeetingsByTime() ในการกรองข้อมูลเพิ่มเติมด้วยการเปรียบเทียบ Timestamp เพื่อให้ได้เฉพาะห้องประชุมที่อยู่ในช่วงเวลาที่กำหนด ใช้สำหรับตรวจสอบความพร้อมของห้องประชุมและป้องกันการจองซ้ำซ้อน

```
async getBooking() {
  await axios.get(`bookingZoom`, {
    params: {
      roomId: 3,
      startDate: this.start.date + " 00:00:00",
      endDate: this.end.date + " 23:59:59",
    }
  }).then((res) => {
    this.booking = res.data
    this.bookingZoom()
  }).catch((error) => {
    console.log(error)
  })
},
```

ภาพประกอบ 3 คำสั่งเรียกดูข้อมูลห้องประชุมทั้งหมดตามวันและเวลา

คำสั่งเพิ่มใบอนุญาต ภาพแสดงฟังก์ชัน addLicense($email, $licenseType) ที่ใช้ HTTP PATCH Request ไปยัง endpoint /users/{email} เพื่ออัปเกรดบัญชีผู้ใช้จาก Basic (Type 1) เป็น Licensed (Type 2) ทำให้ผู้ใช้สามารถเป็น Host และสร้างห้องประชุมได้ไม่จำกัดเวลา ระบบจะตรวจสอบจำนวนใบอนุญาตที่เหลืออยู่ก่อนดำเนินการ และบันทึก Log การจัดสรรใบอนุญาตเพื่อการติดตามและจัดการทรัพยากรอย่างมีประสิทธิภาพ

```
public function addLicense(Request $request)
{
    return $this->_http->patch($this->zoomUrl . 'users/'.$request->id_zoom, [
        'type' => $request->type,
    ])->json();
}
```

ภาพประกอบ 4 คำสั่งเพิ่มใบอนุญาต

คำสั่งลบห้องประชุม ภาพแสดงฟังก์ชัน deleteMeeting($meetingId) ที่ใช้ HTTP DELETE Request ไปยัง endpoint /meetings/{meetingId} เพื่อลบห้องประชุมออกจากระบบ เมื่อการประชุมเสร็จสิ้นหรือถูกยกเลิก ฟังก์ชันจะ return true หากลบสำเร็จ (HTTP Status 204) ระบบจะอัปเดตสถานะใน Database และส่งการแจ้งเตือนไปยังผู้ที่เกี่ยวข้อง พร้อมทั้งคืนใบอนุญาตหากผู้ใช้ไม่มีการประชุมอื่นที่กำลังจะมาถึง

```
public function deleteMeeting($meeting_room_id)
{
    return $this->_http->delete($this->zoomUrl . 'meetings/'.$meeting_room_id)->json();
}
```

ภาพประกอบ 5 คำสั่งลบห้องประชุม

ขั้นตอนที่ 5 การพัฒนาและทดสอบแบบทีละขั้น (Iterative Development and Testing)

การพัฒนาระบบได้ดำเนินการโดยใช้วิธีการพัฒนาทีละขั้น (Iterative Development) ซึ่งช่วยให้สามารถปรับปรุงและแก้ไขปัญหาได้อย่างต่อเนื่อง ในแต่ละรอบการพัฒนา จะมีการทดสอบระบบเพื่อตรวจสอบความถูกต้องและประสิทธิภาพของฟังก์ชันต่าง ๆ เมื่อการพัฒนาทุกรอบเสร็จสิ้น ได้ดำเนินการทดสอบระบบรวม (Integration Testing) เพื่อให้มั่นใจว่าส่วนต่าง ๆ ของระบบสามารถทำงานร่วมกันได้อย่างราบรื่น ข้อบกพร่องที่พบระหว่างการทดสอบได้รับการแก้ไข และปรับปรุงอย่างทันที

ขั้นตอนที่ 6 การทดสอบระบบครบวงจร (Comprehensive System Testing)

เมื่อการพัฒนาฟีเจอร์ทั้งหมดเสร็จสิ้น ได้ดำเนินการทดสอบระบบในรูปแบบต่าง ๆ อย่างครอบคลุม โดยเริ่มจากการทดสอบประสิทธิภาพ (Performance Testing) ด้วยเครื่องมือเจเมเตอร์ (JMeter) เพื่อประเมินความสามารถในการรองรับผู้ใช้งานพร้อมกัน ทดสอบกับผู้ใช้งานพร้อมกัน (Concurrent Users) 100 300 500 และ 700 ราย ตามด้วยการทดสอบการใช้งาน (Usability Testing) กับผู้ใช้งานจริง 30 ราย โดยใช้มาตรฐานการใช้งานระบบ (System Usability Scale - SUS) และอัตราการทำงานสำเร็จ (Task Completion Rate) เพื่อประเมินความง่ายในการใช้งาน สุดท้ายคือการทดสอบระบบรักษาความปลอดภัย (Security Testing) โดยใช้เครื่องมือโอวาสป์แซป (OWASP ZAP) เพื่อตรวจสอบช่องโหว่ทางความปลอดภัย รวมถึงการโจมตีแบบฉีดโซ่เอสคิวแอล (SQL Injection) การโจมตีสคริปต์ข้ามเว็บไซต์ (XSS Attacks) และการปลอมแปลงคำขอข้ามเว็บไซต์ (CSRF Attacks)

ขั้นตอนที่ 7 การติดตั้งและปรับใช้ (Implementation and Deployment)

เมื่อระบบผ่านการทดสอบทั้งหมดแล้ว ได้เริ่มต้นกระบวนการติดตั้งระบบบนเซิร์ฟเวอร์และเตรียมพร้อมสำหรับการใช้งานจริง การปรับใช้ระบบได้ดำเนินการแบบเป็นขั้นตอน (Phased Implementation) เพื่อลดความเสี่ยงและให้สามารถปรับแก้ปัญหาได้ทันท่วงที โดยเริ่มจากการทดลองใช้กับกลุ่มทดสอบจำนวน 30 ราย เพื่อสังเกตการณ์และรับข้อเสนอแนะเบื้องต้น หลังจากที่ระบบทำงานได้อย่างเสถียรและได้รับการปรับปรุงตามข้อเสนอแนะแล้ว จึงได้ขยายการใช้งานไปยังผู้ใช้งานทั้งหมด ระหว่างการดำเนินการนี้ ได้มีการติดตามและประเมินผลการใช้งานอย่างต่อเนื่อง พร้อมทั้งจัดทำเอกสารคู่มือการใช้งานเพื่อช่วยเหลือผู้ใช้งานในการทำความเข้าใจระบบใหม่

การเก็บรวบรวมข้อมูล

การพัฒนาระบบได้ดำเนินการสำรวจความต้องการจากกลุ่มตัวอย่างที่คัดเลือกจากประชากรเป้าหมาย ซึ่งเป็นบุคลากรของมหาวิทยาลัยราชภัฏยะลาทั้งหมด จำนวน 832 คน ประกอบด้วยอาจารย์ เจ้าหน้าที่สายสนับสนุน และบุคลากรสายบริหาร โดยใช้วิธีการสุ่มตัวอย่างแบบเจาะจง คัดเลือกผู้ใช้งานจำนวน 50 ราย ตามเกณฑ์การคัดเลือกที่กำหนด ได้แก่ บุคลากรที่มีประสบการณ์การใช้งานแพลตฟอร์มการประชุมออนไลน์อย่างสม่ำเสมอ มีความต้องการใช้งานใบอนุญาต Zoom เป็นประจำในการปฏิบัติงาน และเป็นตัวแทนจากทุกหน่วยงานภายในมหาวิทยาลัย เพื่อให้ได้ข้อมูลที่ครอบคลุม และ

แม่นยำจากทุกกลุ่มผู้ใช้งาน โดยใช้วิธีการสัมภาษณ์เชิงลึก ร่วมกับแบบสอบถามที่ออกแบบมาเป็นพิเศษ การกระจายตัวของกลุ่มตัวอย่างประกอบด้วย อาจารย์ 25 ราย (ร้อยละ 50) เจ้าหน้าที่สายสนับสนุน 15 ราย (ร้อยละ 30) และบุคลากรสายบริหาร 10 ราย (ร้อยละ 20) เพื่อให้สะท้อนสัดส่วนการใช้งานจริงและความต้องการที่หลากหลายของแต่ละกลุ่ม จากการวิเคราะห์ข้อมูลที่ได้รับ พบว่า ปัญหาหลักที่ผู้ใช้งานประสบคือกระบวนการจองใบอนุญาตที่ซับซ้อน และขาดประสิทธิภาพ ทำให้เกิดความล่าช้าและความไม่สะดวกในการใช้งาน ผลจากการวิเคราะห์นี้ถูกนำมาจัดทำเป็นเอกสารความต้องการของระบบ ที่ชัดเจนและสามารถนำไปใช้ในขั้นตอนการออกแบบต่อไป

การวิเคราะห์ข้อมูล

การวิเคราะห์ข้อมูล (Data Analysis) ใช้วิธีการแบบผสมผสาน (Mixed-Method Analysis) ที่ผสมผสานการวิเคราะห์ข้อมูลเชิงปริมาณ (Quantitative Analysis) และเชิงคุณภาพ (Qualitative Analysis) โดยแบ่งออกเป็นขั้นตอนต่าง ๆ ตามลำดับการดำเนินการวิจัย (Research Process)

การวิเคราะห์ข้อมูลความต้องการ (Requirements Analysis) จากกลุ่มตัวอย่าง (Sample Group) 50 ราย ใช้สถิติเชิงพรรณนา (Descriptive Statistics) สำหรับข้อมูลจากแบบสอบถาม (Questionnaire Data) และการวิเคราะห์เนื้อหา (Content Analysis) กับการจัดหมวดหมู่ประเด็น (Thematic Analysis) สำหรับข้อมูลจากการสัมภาษณ์เชิงลึก (In-depth Interview Data)

การวิเคราะห์ข้อมูลการทดสอบประสิทธิภาพ (Performance Testing Analysis) ใช้ข้อมูลจาก Apache JMeter ที่ทดสอบกับผู้ใช้งานพร้อมกัน (Concurrent Users) 100 300 500 และ 700 ราย โดยวิเคราะห์เวลาตอบสนอง (Response Time) อัตราความเร็วในการประมวลผล (Throughput) และอัตราการเกิดข้อผิดพลาด (Error Rate) ด้วยสถิติเชิงพรรณนา (Descriptive Statistics)

การวิเคราะห์ข้อมูลการทดสอบการใช้งาน (Usability Testing Analysis) กับผู้ใช้งานจริง (Real Users) 30 ราย ใช้มาตราฐาน System Usability Scale (SUS) อัตราการทำงานสำเร็จ (Task Completion Rate) และเวลาเฉลี่ยที่ใช้ในการทำงาน (Task Completion Time) การทดสอบความปลอดภัย (Security Testing) ใช้OWASP ZAP โดยจัดหมวดหมู่ช่องโหว่ (Vulnerability Classification) ตามระดับความรุนแรง (Severity Level) และประเภทการโจมตี (Attack Types)

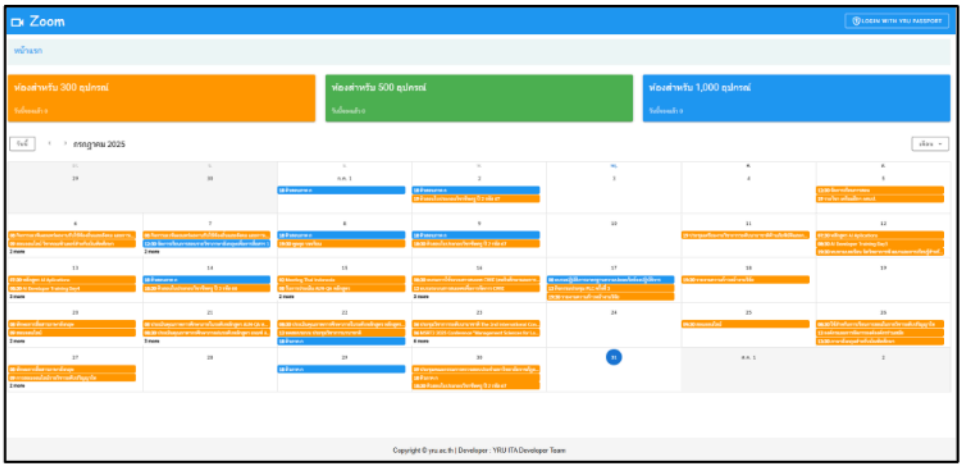
การวิเคราะห์ข้อมูลจากการทดลองใช้งานจริง (Pilot Implementation Analysis) กับกลุ่มทดสอบ (Test Group) 30 ราย เป็นระยะเวลา 4 สัปดาห์ ใช้การวิเคราะห์รูปแบบการใช้งาน (Usage Pattern Analysis) จากข้อมูล Log Files การวิเคราะห์แนวโน้ม (Trend Analysis) และการวิเคราะห์เนื้อหาจากการสัมภาษณ์กลุ่ม (Focus Group Interview Analysis)

การวิเคราะห์ข้อมูลแบบผสมผสาน (Triangulation Analysis) ดำเนินการเพื่อตรวจสอบความสอดคล้องของข้อมูล (Data Consistency) และการวิเคราะห์ความสัมพันธ์ (Correlation Analysis) ระหว่างตัวแปรต่าง ๆ (Variables) รวมถึงการประเมินประสิทธิภาพ (Performance Evaluation) โดยรวมด้วยการคำนวณอัตราการใช้งานระบบ (System Utilization Rate) การประเมินผลตอบแทนจากการลงทุน (Return on Investment: ROI) และการวิเคราะห์ต้นทุนผลประโยชน์ (Cost-Benefit Analysis)

ผลการวิจัย

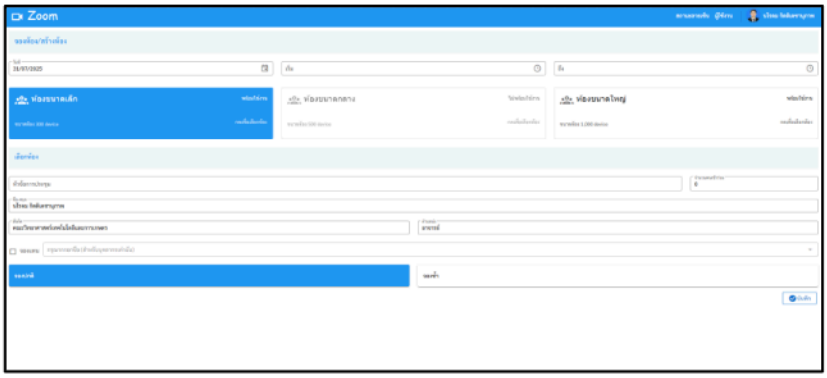
ผลการพัฒนาระบบ

ระบบมีฟีเจอร์หลักที่ประกอบด้วย ระบบจองใบอนุญาตแบบเรียลไทม์ที่ผู้ใช้งานสามารถจองใบอนุญาตได้ทันทีผ่านเว็บไซต์ ระบบจัดการใบอนุญาตอัตโนมัติที่สามารถเพิ่ม และลบใบอนุญาตตามเวลาที่กำหนดได้อัตโนมัติ และระบบรายงานการใช้งานที่แสดงสถิติการใช้งานแบบเรียลไทม์ โดยสามารถเข้าใช้งานผ่านเว็บบราวเซอร์ (Web browser) ตัวอย่าง หน้าแรกของระบบจองห้องประชุมออนไลน์ ระบบการจองห้องประชุมออนไลน์ และรายงานสถิติการใช้งานห้องประชุมออนไลน์ ดังภาพประกอบ 6-8 ตามลำดับ ภาพแสดงหน้าแรก (Dashboard) ของระบบจองห้องประชุมออนไลน์ที่ URL: https://zoom.yru.ac.th ประกอบด้วยตารางแสดงสถานะห้องประชุมแบบ Real-time ที่แบ่งตามความจุ (300, 500, 1000 ที่นั่ง) พร้อมระบุสถานะว่าง/ไม่ว่างด้วยสีสัญลักษณ์ มีส่วนค้นหา และกรองตามวันเวลา และแสดงสถิติการใช้งานโดยรวม ผู้ใช้สามารถเข้าสู่ระบบผ่าน Single Sign-On (SSO) และเลือกจองห้องประชุมได้ทันที



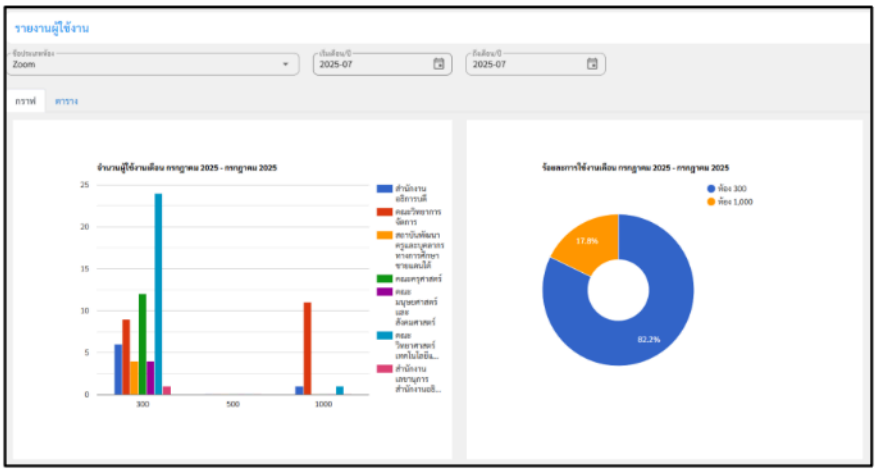
ภาพประกอบ 6 หน้าแรกของระบบจองห้องประชุมออนไลน์

ภาพแสดงการจองห้องประชุม ประกอบด้วยการเลือกประเภทห้องประชุม กรอกรายละเอียด (หัวเรื่อง ผู้จัดการประชุม หน่วยงาน) เลือกวันเวลา และยืนยันการจอง ระบบมีการตรวจสอบความพร้อมแบบ Real-time และแจ้งเตือนหากมีการจองซ้ำซ้อน



ภาพประกอบ 7 หน้าจองระบบจองห้องประชุมออนไลน์

ภาพแสดงหน้า Dashboard Analytics ที่แสดงสถิติการใช้งานแบบ Real-time ประกอบด้วย กราฟแสดงจำนวนการจองแยกตามประเภทห้อง แนวโน้มการใช้งานตามเวลา สัดส่วนการใช้งานตามหน่วยงาน และตารางข้อมูลการจองล่าสุด รวมถึงช่วงเวลาที่มีการใช้งานสูงสุด (Peak Hours) ข้อมูลสามารถกรองตามช่วงวันที่และส่งออกเป็น PDF, Excel, CSV เพื่อใช้ในการวิเคราะห์และวางแผนทรัพยากรในอนาคต



ภาพประกอบ 8 หน้าจองรายงานสถิติการใช้งานห้องประชุมออนไลน์

ระบบมีส่วนติดต่อผู้ใช้งานที่เข้าใจง่ายและใช้งานสะดวก ผู้ใช้งานสามารถเข้าสู่ระบบด้วยระบบลงชื่อเข้าใช้เดียว (Single Sign-On - SSO) ผ่านพาสปอร์ต (Passport) ของมหาวิทยาลัย และสามารถจองห้องประชุมได้ในขั้นตอนที่ง่ายดาย หน้าจอหลักของระบบได้รับการออกแบบให้มีความเรียบง่าย เข้าใจง่าย และใช้งานได้อย่างมีประสิทธิภาพ

ผลการประเมินประสิทธิภาพ

ด้านการใช้งาน

ผลการเปรียบเทียบ เวลาการจอง จำนวนขั้นตอนการจอง ความรวดเร็วในการดำเนินการ ความสะดวกในการใช้งาน ข้อมูลดังตารางที่ 1

ตารางที่ 1 การเปรียบเทียบเวลาและขั้นตอนการจองก่อนและหลังใช้ระบบ

ตัวชี้วัด	ก่อนใช้ระบบ	หลังใช้ระบบ	ผลต่าง	ร้อยละการปรับปรุง	ระดับการปรับปรุง
เวลาการจอง (นาที)	30	3	-27	90.00	ดีมาก
จำนวนขั้นตอนการจอง	10	4	-6	60.00	ดีมาก
ความรวดเร็วในการดำเนินการ	ช้า	รวดเร็ว	เพิ่มขึ้น 10 เท่า	-	ดีมาก
ความสะดวกในการใช้งาน	ต่ำ	สูง	-	-	ปรับปรุงอย่างมีนัยสำคัญ

ด้านประสิทธิภาพระบบ

ผลการเปรียบเทียบประสิทธิภาพ 3 ด้าน ได้แก่ การรองรับผู้ใช้งาน ความเร็ว ความเสถียร และความถูกต้อง ดังตารางที่ 2

ตารางที่ 2 การเปรียบเทียบประสิทธิภาพก่อนและหลังใช้ระบบ

ตัวชี้วัด	ก่อนใช้ระบบ	หลังใช้ระบบ	การปรับปรุง
การรองรับผู้ใช้งาน			
จำนวนผู้ใช้งานพร้อมกัน	จำกัด/ไม่ระบุ	500 ราย	เพิ่มขึ้นอย่างมีนัยสำคัญ
ระบบจัดการคิว	ไม่มี	มีระบบจัดการคิวอัตโนมัติ	ปรับปรุงการจัดสรรทรัพยากร
ความเร็ว			
เวลาตอบสนอง	Manual/ช้า	1.20 วินาที	รวดเร็วทันที
การประมวลผล	Manual	อัตโนมัติแบบ Real-time	เร็วขึ้น 25 เท่า
ความเสถียร			
ความพร้อมใช้งาน	ไม่แน่นอน	99.80% (24/7)	พร้อมใช้ตลอดเวลา
การหยุดทำงาน	บ่อยครั้ง	1.44 ชั่วโมง/เดือน	ลดลงอย่างมาก
ความถูกต้อง			
อัตราข้อผิดพลาด	สูง (Manual Error)	< 0.01%	ลดลง 99%+
ความถูกต้องของข้อมูล	ต่ำ	99.99%	เพิ่มขึ้นอย่างมีนัยสำคัญ

ผลการประเมินความพึงพอใจ

จากการสำรวจความพึงพอใจของผู้ใช้งาน 256 คน โดยใช้แบบสอบถามมาตรฐาน 5 ระดับ พบว่าผู้ใช้งานมีความพึงพอใจในระดับดีมาก โดยมีคะแนนเฉลี่ยโดยรวม 4.20 คะแนน ดังตารางที่ 3

ตารางที่ 3 ผลการประเมินความพึงพอใจของผู้ใช้งานจำแนกตามด้าน

ด้านการประเมิน	คะแนนเฉลี่ย	S.D.	ร้อยละ	ระดับความพึงพอใจ
ความง่ายในการใช้งาน	4.30	0.65	86.00	ดีมาก
ความเร็วในการตอบสนอง	4.20	0.72	84.00	ดีมาก
ความถูกต้องของข้อมูล	4.40	0.58	88.00	ดีมาก
ความพึงพอใจโดยรวม	4.20	0.68	84.00	ดีมาก

อภิปรายผลการวิจัย

การพัฒนาระบบการจองห้องประชุมออนไลน์ประสบความสำเร็จตามวัตถุประสงค์ที่กำหนดไว้ โดยสามารถปรับปรุงประสิทธิภาพการจัดการทรัพยากรและลดต้นทุนดำเนินงานได้อย่างมีนัยสำคัญ ผลการศึกษาพบว่าระบบการจองแบบรวมศูนย์ช่วยเพิ่มประสิทธิภาพการใช้งานทรัพยากร ซึ่งสอดคล้องกับการศึกษาของ Thompson และ Johnson (2020) ที่รายงานว่า การจัดการทรัพยากรแบบรวมศูนย์สามารถเพิ่มประสิทธิภาพได้ร้อยละ 50-85 การปรับปรุงดังกล่าวเกิดจากการลดความซ้ำซ้อนในการจอง การป้องกันการจองที่ขัดแย้งกัน และการใช้ประโยชน์จากทรัพยากรอย่างเต็มศักยภาพ

การบูรณาการ (Integration) ระบบการจองกับระบบการประชุมออนไลน์ (Online Meeting System) ส่งผลให้การดำเนินงานมีประสิทธิภาพสูงขึ้น ผู้ใช้งาน (User) สามารถจัดการการประชุมแบบครบวงจรในระบบเดียว ทำให้ลดเวลาในการเตรียมการและเพิ่มความสะดวกในการใช้งาน การเลือกใช้สถาปัตยกรรม (Architecture) RESTful API เป็นแนวทางที่เหมาะสมและสอดคล้องกับมาตรฐานสากล ตามที่ Rodriguez และ Martinez (2021) ได้ชี้ให้เห็นว่า RESTful API เป็นรูปแบบที่เหมาะสำหรับระบบที่ต้องการความยืดหยุ่นและประสิทธิภาพสูง โครงสร้างแบบไมโครเซอร์วิส (Microservices) ทำให้การพัฒนาและบำรุงรักษาระบบแต่ละส่วนสามารถดำเนินการได้อย่างอิสระ ส่งผลให้การปรับปรุงระบบในอนาคตมีความยืดหยุ่นและมีความเสี่ยงต่ำ

ระบบจองห้องประชุมออนไลน์สร้างผลกระทบเชิงบวกต่อองค์กรในหลายมิติ ได้แก่ การลดต้นทุนดำเนินงาน การเพิ่มความโปร่งใสในการจัดการทรัพยากร การปรับปรุงประสบการณ์ของผู้ใช้งาน (User Experience) และการสร้างฐานข้อมูลการใช้งาน (Usage Data) ที่สามารถนำไปวิเคราะห์เพื่อการวางแผนทรัพยากรในอนาคต อย่างไรก็ตาม การศึกษานี้พบข้อจำกัดสำคัญที่ต้องพิจารณา ได้แก่ การพึ่งพาการเชื่อมต่ออินเทอร์เน็ต (Internet Connection) และส่วนต่อประสานโปรแกรมประยุกต์

(Application Program Interface: API) ของ Zoom การพึ่งพา ส่วนต่อประสานโปรแกรมประยุกต์ (Application Program Interface: API) ของบุคคลที่สาม (Third-party API) ก่อให้เกิดความเสี่ยงจากการเปลี่ยนแปลงนโยบายหรือการอัพเดต (Update) ของผู้ให้บริการ ซึ่งองค์กรไม่สามารถควบคุมได้โดยตรง

ความท้าทายด้านความปลอดภัยของข้อมูล (Data Security) เป็นประเด็นสำคัญที่ต้องให้ความสนใจ การเก็บข้อมูลการจอง (Booking Data) และการเชื่อมต่อกับระบบภายนอกต้องมีการป้องกันที่เหมาะสม รวมถึงการควบคุมการเข้าถึง (Access Control) อย่างเข้มงวดเพื่อป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต (Unauthorized Access) และการรั่วไหลของข้อมูล (Data Breach) นอกจากนี้ การใช้บริการของบุคคลที่สาม (Third-party Service) มีข้อจำกัดในการปรับแต่งฟีเจอร์ (Feature Customization) ให้ตรงกับความต้องการเฉพาะขององค์กร ซึ่งอาจส่งผลต่อการตอบสนองความต้องการที่ซับซ้อนหรือเฉพาะเจาะจง

ข้อเสนอแนะ

เพื่อเสริมสร้างความมั่นคงและประสิทธิภาพของระบบ เสนอแนะให้พัฒนาระบบสำรอง (Backup System) หรือใช้กลยุทธ์ผู้ให้บริการหลายราย (Multi-provider Strategy) เพื่อลดความเสี่ยงจากการพึ่งพาเทคโนโลยีภายนอก การพัฒนาระบบการรักษาความปลอดภัย (Security System) ที่แข็งแกร่งควรประกอบด้วยการเข้ารหัสข้อมูล (Data Encryption) การตรวจสอบสิทธิ์แบบหลายชั้น (Multi-factor Authentication) และระบบติดตามการใช้งาน (System Monitoring) แบบเรียลไทม์ ควรพิจารณาการพัฒนาฟีเจอร์เพิ่มเติม เช่น ระบบการแจ้งเตือนอัจฉริยะ (Smart Notification System) การวิเคราะห์รูปแบบการใช้งาน (Usage Pattern Analysis) และการบูรณาการกับระบบอื่นขององค์กร (System Integration) เพื่อเพิ่มประสิทธิภาพและความสะดวกสบาย เช่นระบบนัดหมายการประชุม ระบบการเรียนการสอนออนไลน์

การวิจัยในอนาคตควรศึกษาผลกระทบระยะยาวของการใช้ระบบต่อประสิทธิภาพการทำงานขององค์กร การเปรียบเทียบต้นทุนและผลประโยชน์ในระยะยาว และการพัฒนาแนวทางการประเมินความสำเร็จของระบบการจองแบบรวมศูนย์ นอกจากนี้ ควรศึกษาการประยุกต์ใช้เทคโนโลยีใหม่ เช่น ปัญญาประดิษฐ์ (Artificial Intelligence) และการเรียนรู้ของเครื่อง (Machine Learning) เพื่อพัฒนาความสามารถในการคาดการณ์และปรับปรุงการจัดการทรัพยากรอัตโนมัติ

เอกสารอ้างอิง

Anderson, M., Thompson, R., & Davis, L. (2020). Comparative analysis of PHP frameworks for enterprise resource management systems. *Journal of Web Engineering*, *19*(3), 215-238. doi:10.13052/jwe1540-9589.1936.

Chen, W., Liu, J., & Zhang, H. (2020). The impact of COVID-19 on digital transformation in higher education institutions. *Computers & Education*, *162*, 104-115. doi:10.1016/j.compedu.2020.104115.

Chopin, S., Poirier, A., & Krutskikh, A. (2020). Modern web application development with Nuxt.js: A comprehensive analysis. *IEEE Software*, *37*(4), 45-52. doi:10.1109/MS.2020.2991071.

Fielding, R.T. (2000). *Architectural styles and the design of network-based software architectures* (Unpublished doctoral dissertation). California: University of California, Irvine.

Kumar, A., Patel, S., & Sharma, N. (2019). Centralized resource management in cloud computing environments: A systematic review. *Future Generation Computer Systems*, *97*, 445-458. doi: 10.1016/j.future.2019.03.005.

Lee, S., & Kim, J. (2021). Performance comparison of modern JavaScript frameworks for enterprise applications. *ACM Computing Surveys*, *54*(2), 1-28. doi:10.1145/3447556.

Martinez, E., & Johnson, A. (2020). Digital transformation strategies in educational institutions during pandemic disruption. International Journal of Educational Technology, 24(2), 156-174. doi:10.1016/j.ijedt.2020.03.021.

Mystakidis, S. (2022). Metaverse. *Encyclopedia*, *2*(1), 486-497. doi:10.3390/encyclopedia2010031

Otwell, T. (2021). *Laravel: The definitive guide to building web applications*. New York: Apress.

Rodriguez, C., & Martinez, E. (2021). RESTful API design patterns for scalable web services. *IEEE Internet Computing*, *25*(3), 32-39. doi:10.1109/MIC.2021.3067595.

Thompson, J., & Johnson, A. (2020). Resource pooling strategies in educational technology management. *Educational Technology Research and Development*, *68*(4), 1789-1808. doi:10.1007/s11423-020-09781-9.

Wang, X., Chen, Y., & Li, M. (2021). Online learning platforms adoption during COVID-19: A technology acceptance model perspective. *Computers & Education*, *168*, 104-118. doi:10.1016/j.compedu.2021.104118.