

Received: 21 เม.ย. 2568

Revised: 16 มิ.ย. 2568

Accepted: 24 มิ.ย. 2568

การแจ้งเตือนเพื่อการต่อสัญญาการบำรุงรักษาซอฟต์แวร์โดยใช้การแจ้งเตือนหลายระยะ
Software Maintenance Contract Renewal Notifications using Multi-phase
Reminders

สุจิตรา ปานการะเกตุ¹, สุธิชานนท์ พูลการชาย¹, พาสน์ ปราโมกษ์ชน¹, อลงกต กองมณี¹
และ สมนึก สินธุพาน^{1*}

¹สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้

Sujittra Pankaragat¹, Suthichanon Poonkrankay¹, Part Pramokchon¹,
Alongkot Gongmanee¹ and Somnuek Sinthupuan^{1*}

¹Computer Science Department, Faculty of Science, Maejo University

*Corresponding author: somnuk@mju.ac.th

Abstract

The research study focused on developing an integrated maintenance contract management system with multi-phase customer notification capabilities to facilitate timely contract renewals. By implementing a centralized platform that sends automated alerts at three strategic intervals (30 days, 15 days, and 7 days before expiration), the system achieved a 39.39% overall renewal rate from 33 expiring contracts, successfully renewing 13 contracts. Analysis of the phased notification approach revealed progressively effective results: the initial 30-day alert secured 7 renewals from 17 contracts (41.18%), the 15-day follow-up converted 4 of 10 remaining contracts (40%), and the final 7-day reminder yielded 2 renewals from 6 contracts (33.33%), cumulatively reaching 76.47% of potential renewals. Performance varied significantly across client companies, with Company 1 demonstrating perfect renewal compliance (100% for 3 contracts), Company 4 achieving 66.67% success, Companies 2 and 5 maintaining moderate 33.33% rates, and Company 3 trailing at 22.22%. Although the multi-stage notification strategy substantially improved renewal opportunities, the 23.53% non-renewal rate suggests the need for further analysis of contract terms and customer responsiveness to optimize future retention strategies.

Keywords: Maintenance; Notification; Renewal Contract; Inquiry and Issue Management

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์ เพื่อพัฒนาระบบการจัดการข้อมูลสัญญาบำรุงรักษา และการแจ้งเตือนลูกค้าก่อนวันหมดอายุสัญญาล่วงหน้า ที่สามารถรวบรวมและจัดเก็บข้อมูลทั้งหมดในระบบกลาง ทำให้สามารถดำเนินการต่ออายุสัญญาลูกค้าได้ทันเวลาโดยใช้การแจ้งเตือนหลายระยะ ผลการวิจัยพบว่า การแจ้งเตือนทั้งสามระยะ (เตือนครั้งที่ 1 ภายใน 30 วัน, ครั้งที่ 2 ภายใน 15 วัน, และครั้งที่ 3 ภายใน 7 วัน) มีสัญญาครบกำหนดรวม 33 สัญญา และต่อสัญญาสำเร็จ 13 สัญญา คิดเป็นอัตราความสำเร็จเฉลี่ย 39.39% การเตือนครั้งที่ 1 ต่อสัญญาได้ 7 จาก 17 สัญญา (41.18%) ครั้งที่ 2 ต่อสัญญา 4 จาก 10 สัญญา (40%) และครั้งที่ 3 ต่อสัญญา 2 จาก 6 สัญญา (33.33%) ทำให้ความสำเร็จสะสมสูงสุด 76.47% บริษัท 1 มีประสิทธิภาพสูงสุด (100%) ต่อสัญญาครบทั้ง 3 สัญญา รองลงมาคือบริษัท 4 (66.67%) ส่วนบริษัท 3 ต่ำสุด (22.22%) บริษัท 2 และ 5 มีอัตราความสำเร็จปานกลาง (33.33%) การแจ้งเตือนหลายรอบช่วยเพิ่มโอกาสต่อสัญญา แต่สัญญา 23.53% ยังไม่ต่อ อาจต้องวิเคราะห์เงื่อนไขสัญญาหรือการตอบสนองของลูกค้าเพื่อปรับปรุงกลยุทธ์ต่อไป

คำสำคัญ: การบำรุงรักษา; การแจ้งเตือน; การสัญญาใหม่; การจัดการคำถามและปัญหา

1. บทนำ

ที่มาเนื่องจากในระบบการจัดการซอฟต์แวร์แบบ ธุรกิจกับธุรกิจ (B2B) หรือ ซอฟต์แวร์ในรูปแบบบริการ (Software as a Service) ที่มีการให้บริการหลังการขายแบบมีสัญญา (Maintenance Contract) การจัดการรอบอายุสัญญาเป็นสิ่งสำคัญ เนื่องจากการแจ้งเตือนเพื่อการต่อสัญญาการบำรุงรักษาซอฟต์แวร์ มีอายุจำกัด เช่น 3 เดือน, 6 เดือน หรือ 1 ปี หากไม่ได้ต่ออายุทันเวลา อาจทำให้ระบบไม่ได้รับการบำรุงรักษา (Maintenance Downtime) ลูกค้ารู้สึกว่าจะไม่ได้รับบริการอย่างต่อเนื่อง บริษัทสูญเสียรายได้และความเชื่อมั่นของลูกค้า และหลายองค์กรยังใช้ วิธีการแจ้งเตือนแบบแมนนวล (Manual) ผ่าน Excel, Email หรือโทรศัพท์ ซึ่งมีความเสี่ยงสูงที่จะพลาดกำหนดเวลาดังนั้นจึงเกิดปัญหาหลัก (Problems) ประกอบด้วย 1) การแจ้งเตือนล่าช้า หรือไม่สม่ำเสมอ เนื่องจากระบบอัตโนมัติในการตรวจสอบวันหมดอายุ จึงเกิดการแจ้งล่าช้าหรือไม่แจ้งเลย 2) ไม่มีระบบรวมศูนย์ข้อมูลสัญญา ข้อมูลลูกค้าและสัญญาถูกจัดเก็บแยกกันในหลายไฟล์หรือหลายระบบ ทำให้ยากต่อการติดตาม 3) การสื่อสารไม่ชัดเจนหรือซ้ำซ้อน การแจ้งเตือนอาจไม่ระบุข้อมูลครบ เช่น หมายเลขสัญญา วันหมดอายุ หรือขั้นตอนต่ออายุ ทำให้ลูกค้าเพิกเฉย 4) ลูกค้ามองข้ามการแจ้งเตือน แจ้งเตือนในเวลาหรือช่องทางที่ไม่เหมาะสม เช่น ส่งอีเมลในวันหยุด หรือใช้ช่องทางที่ลูกค้าไม่ตอบสนอง เช่น ระบบภายในแทนที่จะเป็น LINE หรือ SMS และ 5) ไม่มีการวิเคราะห์ผลลัพธ์ของการแจ้งเตือน

ไม่ได้ติดตามว่า การแจ้งเตือนแต่ละครั้งทำให้ลูกค้าต่ออายุหรือไม่ ทำให้ขาดข้อมูลย้อนกลับ (feedback loop) ประกอบกับสัญญาบำรุงรักษามีเงื่อนไขคือ 1) ขอบเขตการให้บริการ การบำรุงรักษาเชิงแก้ไข (Corrective Maintenance) คือ การแก้ไขข้อผิดพลาดที่เกิดขึ้นกับซอฟต์แวร์หลังจากใช้งานไปแล้ว เพื่อให้ซอฟต์แวร์กลับมาทำงานได้อย่างถูกต้อง การบำรุงรักษาเชิงป้องกัน (Preventive Maintenance) คือ การบำรุงรักษาซอฟต์แวร์เพื่อป้องกันปัญหาที่อาจเกิดขึ้นในอนาคต เช่น การตรวจสอบระบบ การปรับปรุงประสิทธิภาพ การบำรุงรักษาเชิงปรับปรุง (Perfective Maintenance) คือการปรับปรุงซอฟต์แวร์ให้มีประสิทธิภาพสูงขึ้น หรือเพิ่มฟังก์ชันการทำงานใหม่ๆ ให้กับซอฟต์แวร์ การสนับสนุนด้านเทคนิค (Technical Support) คือการให้คำปรึกษาและแก้ไขปัญหาที่เกี่ยวข้องกับการใช้งานซอฟต์แวร์ให้กับลูกค้า 2) ระยะเวลาของสัญญา โดยทั่วไปแล้ว สัญญาบำรุงรักษาระบบซอฟต์แวร์จะมีระยะเวลา 1 ปี ภายในบริษัทมีระยะเวลาระยะสั้น เช่น 3 เดือน 6 เดือน และสามารถต่ออายุสัญญาได้เมื่อครบกำหนด 3) ค่าบริการ ค่าบริการในการบำรุงรักษาระบบซอฟต์แวร์จะแตกต่างกันไป ขึ้นอยู่กับขอบเขตการให้บริการและเงื่อนไขของแต่ละสัญญา และ 4) ข้อตกลงและเงื่อนไขอื่นๆ รายละเอียดเกี่ยวกับความรับผิดชอบของผู้ให้บริการและลูกค้า เงื่อนไขการยกเลิกสัญญา ข้อกำหนดเกี่ยวกับความลับของข้อมูล ข้อควรระวัง ควรอ่านและทำความเข้าใจในรายละเอียดของสัญญาบำรุงรักษาระบบซอฟต์แวร์อย่างรอบคอบก่อนที่จะลงนามในสัญญา

ระบบการแจ้งเตือนเพื่อการต่อสัญญาการบำรุงรักษาซอฟต์แวร์โดยใช้การแจ้งเตือนหลายระยะจึงได้ถูกพัฒนาขึ้นโดยใช้เทคโนโลยี React, TypeScript และ Firebase ซึ่งระบบประกอบด้วย ฟังก์ชัน การเข้าสู่ระบบ การจัดการบัญชี การจัดการข้อมูลลูกค้า การจัดการข้อมูลโครงการลูกค้า การจัดการข้อมูลสัญญาบำรุงรักษาโครงการ การแจ้งปัญหา การแก้ไขปัญหาใช้สำหรับการดำเนินการแก้ไขปัญหาที่ได้รับแจ้ง และการจัดทำรายงานที่แสดงสถิติหรือผลการดำเนินงานในระบบ

2. วัตถุประสงค์

เพื่อพัฒนาระบบการจัดการข้อมูลสัญญาบำรุงรักษา และการแจ้งเตือนลูกค้าก่อนวันหมดอายุ สัญญาล่วงหน้าโดยใช้การแจ้งเตือนหลายระยะ (Multi-phase Reminders) 30 วัน 15 วัน และ 7 วัน

3. ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีวงจรชีวิตสัญญา (Contract Lifecycle Management Theory) ทฤษฎีนี้เน้นการบริหารจัดการสัญญาตั้งแต่ต้นจนจบวงจร เริ่มตั้งแต่การร่างและลงนามสัญญา การเก็บรักษาและติดตามสถานะสัญญา รวมถึงการต่ออายุหรือยกเลิกสัญญา หลักการคือการทำให้งานจัดการสัญญา

ประสิทธิภาพ ลดความผิดพลาดและความล่าช้า ระบบแจ้งเตือนจะถูกตั้งค่าให้เตือนเจ้าหน้าที่หรือฝ่ายลูกค้าเมื่อสัญญากำลังจะหมดอายุ เพื่อให้มีเวลาพอในการตัดสินใจต่อสัญญาหรือเจรจาใหม่ ทำให้กระบวนการต่ออายุสัญญารวดเร็วและไม่มีช่องว่างที่อาจเกิดความเสียหายทางธุรกิจ ซึ่งประกอบด้วย 1) การร้องขอสัญญา (Request) 2) การร่างสัญญา (Authoring) 3) การเจรจาเงื่อนไข (Negotiation) 4) การอนุมัติ (Approval) 5) การลงนาม (Execution) 6) การติดตามข้อผูกพัน (Obligation & Compliance) และ 7) การต่ออายุหรือสิ้นสุด (Renewal or Termination) หลักการของทฤษฎี CLM ประกอบด้วย Visibility, Automation, Compliance, Risk Mitigation และ Auditability

ทฤษฎีการสื่อสารและการแจ้งเตือน (Notification Theory) หลักการของทฤษฎีนี้คือการส่งข้อความแจ้งเตือนที่ถูกต้อง ตรงเวลา และผ่านช่องทางที่เหมาะสม เพื่อให้ผู้รับสารสามารถเข้าใจและตอบสนองได้อย่างมีประสิทธิภาพ การแจ้งเตือนที่ดีต้องไม่รบกวนมากเกินไปจนกลายเป็นขยะ (spam) แต่ก็ต้องมีความถี่เพียงพอที่จะกระตุ้นให้ผู้รับปฏิบัติ ผลที่คาดหวังคือเพิ่มอัตราการต่ออายุสัญญา ลดความเสี่ยงที่สัญญาจะหมดอายุโดยไม่รู้ตัว และเพิ่มความพึงพอใจของลูกค้า ประกอบด้วย 1) ความเหมาะสมของเวลา (Timeliness) 2) ความเกี่ยวข้อง (Relevance) 3) ความสามารถในการดำเนินการ (Actionability) 4) ประสิทธิภาพของช่องทาง (Channel Effectiveness) และ 5) ภาระทางความคิด (Cognitive Load) หลักการแจ้งเตือนที่มีประสิทธิภาพควรแจ้งล่วงหน้าแบบมีระยะเวลา เช่น 30, 15 และ 7 วันก่อนหมดอายุ โดยใช้ Multi-phase reminders เพื่อให้ผู้รับมีเวลาพิจารณาและดำเนินการ

4. งานวิจัยที่เกี่ยวข้อง

ณัทยวรรณ วิโสภา สุติเทพ และคณะ (2564) ได้พัฒนาระบบฐานข้อมูลเพื่อจัดการพัสดุของภาควิชาเทคโนโลยีการศึกษา มหาวิทยาลัยเกษตรศาสตร์ โดยศึกษาจากกลุ่มตัวอย่างผู้เชี่ยวชาญ 9 คนและบุคลากร 7 คน ผลการศึกษาพบปัญหาด้านความล่าช้าในการเบิกจ่ายและตรวจสอบพัสดุ ระบบที่พัฒนาขึ้นมีคุณภาพในระดับดีมาก (ค่าเฉลี่ย 4.44, S.D. 0.56) และได้รับความพึงพอใจจากผู้ใช้ในระดับสูง (ค่าเฉลี่ย 4.46, S.D. 0.61) งานวิจัยนี้สอดคล้องกับแนวคิดของ Kumar และคณะ (2024) ที่ศึกษาการประยุกต์ใช้ Agile Methodology ในการบำรุงรักษาซอฟต์แวร์บนคลาวด์ โดยพบว่าการใช้ Continuous Integration/Continuous Deployment (CI/CD) ร่วมกับ Automated Testing ช่วยลดเวลาแก้ไขปัญหาได้ถึง 40% สมนึก สีนรุปวน และคณะ (2568) ได้พัฒนาระบบรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและการแจ้งเตือนผ่านไลน์ เพื่อแก้ไขปัญหาการจัดการปัญหาของลูกค้าที่มีประสิทธิภาพต่ำในบริษัทพัฒนาซอฟต์แวร์ ซึ่งเดิมลูกค้ารายงานปัญหาผ่าน Line หรือ Email ทำให้เกิดความยุ่งยากในการติดตามและแก้ไขปัญหา โดยระบบใหม่นี้ประกอบด้วย 5 องค์ประกอบหลัก ได้แก่ 1) การจัดการข้อมูลค่าคงที่ 2) การรับแจ้งปัญหา 3) การจัดการแก้ปัญหา 4)

การจัดการพนักงาน และ 5) การจัดการโครงการ และใช้วิธีการทดสอบระบบแบบอัตโนมัติด้วยเฟรมเวิร์ค Cypress แทนการทดสอบด้วยมือ ผลการวิจัยพบว่าระบบสามารถทำงานได้ครบถ้วนทุกฟังก์ชันภายในระยะเวลา 6 เดือน ซึ่งเร็วกว่าวิธีเดิม 40% และจากการทดสอบกับบริษัทลูกค้า 2 แห่ง พบว่าระบบสามารถรับรายงานปัญหาได้ทั้งหมด 120 ปัญหา (แบ่งเป็นบริษัทละ 60 ปัญหา) และแก้ไขปัญหาลงมือสำเร็จ 70% และ 84% ตามลำดับ ซึ่งสูงกว่าวิธีเดิมที่แก้ปัญหาลงมือได้เพียง 50% แสดงให้เห็นว่าระบบนี้ช่วยเพิ่มประสิทธิภาพการจัดการปัญหาและลดเวลาในการดำเนินงานได้อย่างมีนัยสำคัญ

ขณะที่ อรรถพล จันทร์สมุด (2563) ที่ใช้ระบบโซลูชันแบบดิจิทัลเพื่อการจัดการโครงการในสถาบันอุดมศึกษาเป็นนวัตกรรมที่นำเทคโนโลยีดิจิทัลมาประยุกต์ใช้เพื่อเพิ่มประสิทธิภาพการบริหารโครงการต่าง ๆ เช่น การเรียนการสอนและการบริการวิชาการ โดยระบบนี้ประกอบด้วย 6 กระบวนการหลัก ได้แก่ 1) ผู้ส่งมอบ 2) ผู้ผลิตโครงการ 3) ลูกค้าโครงการ 4) ผู้บริโภค 5) ความพึงพอใจ และ 6) การคืนกลับ ซึ่งทั้งหมดเชื่อมต่อกันผ่านระบบดิจิทัลแบบเรียลไทม์ ช่วยให้การดำเนินงานมีความรวดเร็ว โปร่งใส และลดข้อผิดพลาด ผลการนำระบบมาใช้พบว่าสามารถลดค่าใช้จ่าย เพิ่มประสิทธิภาพการทำงาน สร้างความพึงพอใจให้ผู้ใช้งาน และพัฒนาคุณภาพการบริการของสถาบันการศึกษาได้อย่างมีนัยสำคัญ Wang และคณะ (2024) วิเคราะห์การใช้ GitHub Actions ในโครงการโอเพนซอร์ส 200 โครงการ พบว่า 47% ของเวิร์กโฟลว์ต้องได้รับการอัปเดตภายใน 6 เดือนเนื่องจาก dependency ที่ล้าสมัย และแนะนำให้ใช้ Dependency Bot เพื่อจัดการปัญหานี้

Fernández และคณะ (2024) ได้ทำการศึกษาวิธีการ Hot Patching ซึ่งเป็นเทคนิคการอัปเดตระบบแบบไม่ต้องปิดการทำงาน เหมาะสำหรับระบบที่ต้องการความต่อเนื่องสูง เช่น ระบบธุรกรรมการเงินหรือระบบควบคุมโรงพยาบาล การศึกษานี้แบ่งเทคนิคออกเป็นสามระดับตามความเสี่ยง โดยพบว่าวิธีการแก้ไขหน่วยความจำขณะระบบทำงาน (Memory Patching) มีประสิทธิภาพสูงสุดแต่ก็มีความเสี่ยงมากที่สุดเช่นกัน ผู้วิจัยแนะนำให้ใช้เทคนิคนี้เฉพาะกับปัญหาที่สำคัญจริงๆ และต้องมีการทดสอบอย่างละเอียดก่อนนำไปใช้งานจริง Chen และคณะ (2024) มุ่งเน้นการศึกษาปัญหาการจ้างงานภายนอก (Outsourcing) ในการพัฒนาระบบซอฟต์แวร์ ผลการศึกษาจาก 50 โครงการพบว่าปัญหาหลักเกิดจากการสื่อสารที่ไม่ชัดเจนและมาตรฐานงานที่ไม่สม่ำเสมอ งานวิจัยนี้เสนอให้ใช้กลไก Service Level Agreement (SLA) ที่ระบุรายละเอียดการทำงานอย่างชัดเจน พร้อมกับกำหนดตัวชี้วัดประสิทธิภาพ (KPIs) ที่สามารถวัดผลได้จริง ซึ่งช่วยลดปัญหาความล่าช้าในการส่งมอบงานได้ถึง 35% และเพิ่มความพึงพอใจของลูกค้า Saito และคณะ (2024) ได้พัฒนาระบบพยากรณ์ความผิดพลาดของซอฟต์แวร์โดยใช้การเรียนรู้ของเครื่อง (Machine Learning) ร่วมกับหลักการ Reliability-Centered Maintenance ระบบนี้สามารถวิเคราะห์ข้อมูลจากบันทึกระบบ (Log Files) ประวัติการซ่อมบำรุง และรูปแบบการใช้งาน เพื่อทำนายจุดที่อาจเกิดปัญหาล่วงหน้า ผลการทดลองแสดงให้เห็นว่าระบบสามารถตรวจจับปัญหาที่อาจเกิดขึ้นได้ล่วงหน้า 72 ชั่วโมงด้วยความแม่นยำ

89% ช่วยให้ทีมงานมีเวลาเตรียมการแก้ไขปัญหอย่างเหมาะสม Brown และคณะ (2023) เสนอกรอบการทำงาน (Framework) ใหม่สำหรับการบำรุงรักษาซอฟต์แวร์ที่เน้นการป้องกันปัญหาตั้งแต่ต้นทาง กรอบการทำงานนี้ประกอบด้วยสามองค์ประกอบหลัก ได้แก่ การตรวจสอบระบบแบบเรียลไทม์ (Real-time Monitoring) การทดสอบระบบตั้งแต่ช่วงออกแบบ (Shift-Left Testing) และระบบรับฟังความคิดเห็นจากผู้ใช้แบบรวดเร็ว (Fast Feedback Loop) การนำกรอบการทำงานนี้ไปใช้ช่วยลดจำนวนปัญหาที่พบในระบบผลิตได้ถึง 40% และลดเวลาแก้ไขปัญหาลดลง 25% Rossi และคณะ (2024) พัฒนาแบบจำลองการประมาณต้นทุนการบำรุงรักษาที่คำนึงถึงปัจจัยด้าน Technical Debt และความซับซ้อนของโค้ด (Code Complexity) แบบจำลองนี้ช่วยให้องค์กรสามารถวางแผนงบประมาณได้อย่างมีประสิทธิภาพมากขึ้น โดยพิจารณาจากความเสี่ยงและผลกระทบที่อาจเกิดขึ้นจากการบำรุงรักษา ผลการทดลองแสดงให้เห็นว่าแบบจำลองนี้สามารถช่วยองค์กรลดค่าใช้จ่ายที่ไม่จำเป็นลงได้เฉลี่ย 15-20% ต่อปี Ahmed และคณะ (2022) วิจัยการใช้แบบจำลองปัญญาประดิษฐ์แบบผสม (Hybrid AI Models) ในการประเมินความง่ายในการบำรุงรักษาซอฟต์แวร์ (Maintainability) ผลการศึกษาพบว่าแบบจำลองที่ผสมผสานระหว่าง Genetic Algorithm และ Fuzzy Logic ให้ผลการประเมินที่แม่นยำที่สุด (ความแม่นยำ 92%) เมื่อเทียบกับวิธีการอื่นๆ การศึกษานี้ช่วยให้นักพัฒนาสามารถออกแบบระบบให้ง่ายต่อการบำรุงรักษาตั้งแต่เริ่มต้น Lim และคณะ (2023) พัฒนารอบการวิเคราะห์ความคุ้มค่า (Cost-Benefit Analysis Framework) เฉพาะสำหรับธุรกิจขนาดกลางและขนาดย่อม (SMEs) กรอบการทำงานนี้ช่วยให้องค์กรขนาดเล็กสามารถจัดลำดับความสำคัญของงานบำรุงรักษาได้อย่างมีประสิทธิภาพ โดยพิจารณาจากผลกระทบทางธุรกิจและทรัพยากรที่มีอยู่ ผลการนำไปใช้จริงแสดงให้เห็นว่าองค์กรสามารถลดค่าใช้จ่ายในการบำรุงรักษาลงได้เฉลี่ย 18% โดยไม่ส่งผลกระทบต่อคุณภาพของระบบ

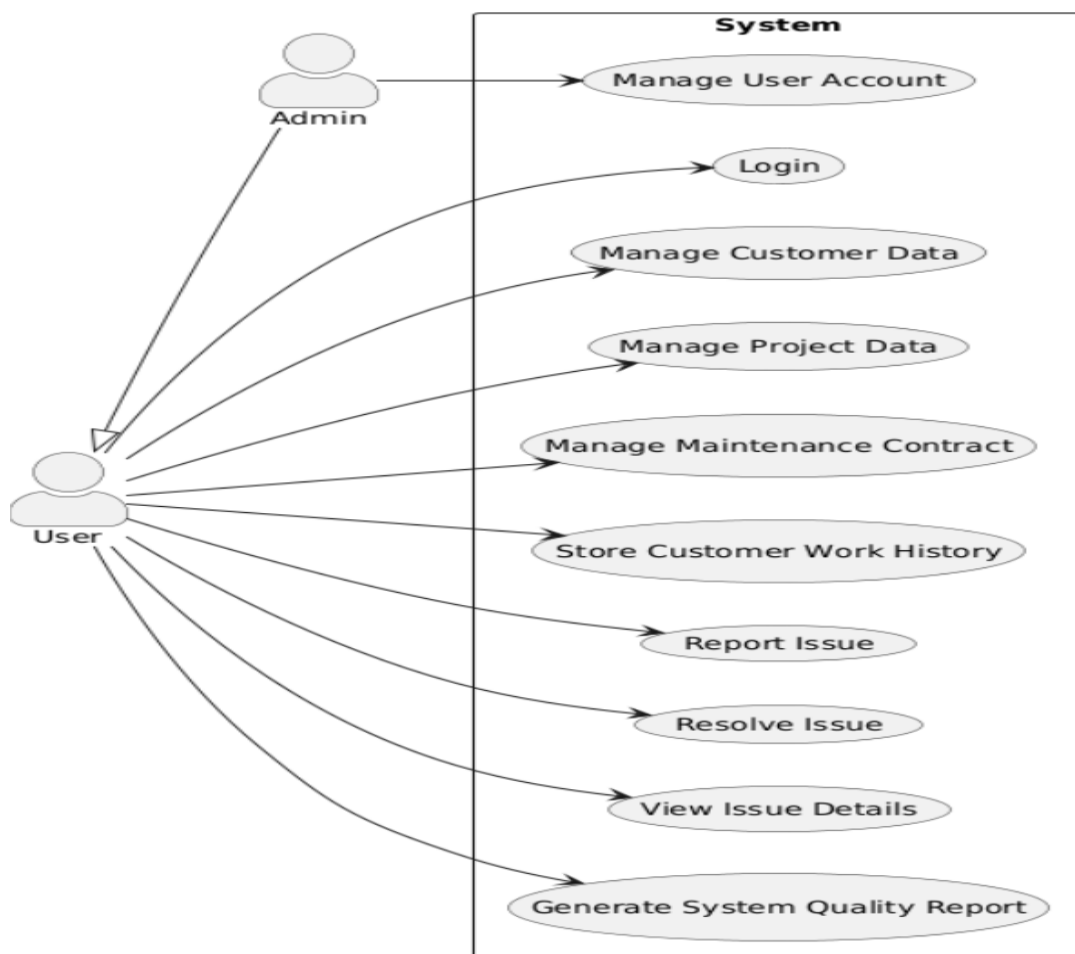
5. สถาปัตยกรรมระบบ (Three-Tier Architecture)

สถาปัตยกรรมแบบ Three-Tier Architecture ซึ่งแบ่งออกเป็น 3 ชั้นหลัก ประกอบด้วย

- 1) **ชั้นการแสดงผลและโต้ตอบกับผู้ใช้งาน (Presentation Layer)** เป็นส่วนที่ติดต่อกับผู้ใช้งานโดยตรง ซึ่งพัฒนาโดยใช้ React ร่วมกับ Vite เพื่อให้ได้ประสิทธิภาพการโหลดที่รวดเร็วและตอบสนองผู้ใช้ได้ดี ผู้ใช้สามารถเข้าถึงข้อมูลสัญญา แจ้งเตือน และดำเนินการต่อสัญญาผ่านอินเทอร์เน็ตได้
- 2) **ชั้นตรรกะของระบบ (Application Layer)** ทำหน้าที่เป็นตัวกลางที่เชื่อมต่อระหว่างหน้าบ้านกับฐานข้อมูล โดยจัดการตรรกะทางธุรกิจต่าง ๆ เช่น การคำนวณวันหมดอายุของสัญญา การส่งการแจ้งเตือนในหลายระยะเวลา (เช่น 30, 15, 7 วันก่อนครบอายุ) การตรวจสอบสิทธิ์ผู้ใช้งาน และการควบคุมกระบวนการต่ออายุสัญญา โดยมักพัฒนาในรูปแบบ RESTful API ด้วย Node.js และ Express หรือเทคโนโลยีที่เทียบเท่า
- 3) **ชั้นฐานข้อมูล (Data Layer)** ซึ่งเป็นชั้นที่เก็บ

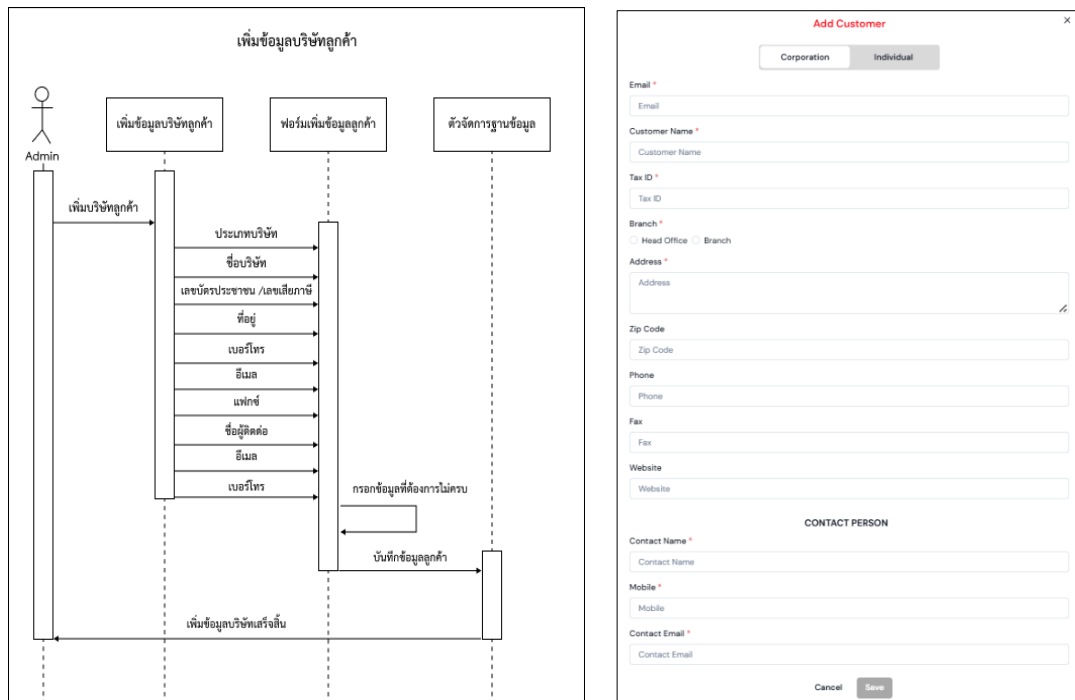
ข้อมูลทั้งหมดของระบบ เช่น ข้อมูลลูกค้า รายละเอียดสัญญา ประวัติการแจ้งเตือน และข้อมูลผู้ใช้งาน โดยใช้ระบบฐานข้อมูลเชิงสัมพันธ์ เช่น MySQL หรือ PostgreSQL

5.1 ชั้นการแสดงผลและโต้ตอบกับผู้ใช้งาน (Presentation Layer) การแสดงผลและโต้ตอบกับผู้ใช้งานทำหน้าที่เป็นหน้าบ้านของระบบ ที่เชื่อมต่อผู้ใช้งานกับฟังก์ชันต่าง ๆ ของระบบผ่านอินเทอร์เน็ตที่ใช้งานง่าย โดยแผนภาพ Use Case Diagram ที่กำหนดขึ้นนี้ แสดงถึงความสัมพันธ์ระหว่างผู้ใช้งานสองประเภท ได้แก่ User และ Admin กับชุดฟังก์ชันที่มีอยู่ภายในระบบ โดยใช้สัญลักษณ์ของ UML อย่างครบถ้วนและถูกต้อง Actor ทั้งสองถูกแทนด้วยรูปคน ซึ่ง User เป็นผู้ใช้งานทั่วไปที่สามารถเข้าถึงเกือบทุกฟังก์ชันของระบบ เช่น การเข้าสู่ระบบ จัดการข้อมูลลูกค้า ข้อมูลโครงการ รายงานปัญหา และดูรายละเอียดปัญหา ในขณะที่ Admin ซึ่งถูกกำหนดให้สืบทอดสิทธิจาก User ผ่านสัญลักษณ์ <|-- สามารถเข้าถึงฟังก์ชันเหล่านั้นได้เช่นกัน รวมถึงมีสิทธิพิเศษเพิ่มเติมในการจัดการบัญชีผู้ใช้งานผ่าน Use Case "Manage User Account" ที่สงวนไว้เฉพาะ Admin ความสัมพันธ์ระหว่าง Actor กับ Use Case ถูกแสดงด้วยลูกศร --> ซึ่งบ่งบอกว่า Actor นั้นสามารถใช้งานฟังก์ชันที่กำหนดได้ ดังภาพที่ 1



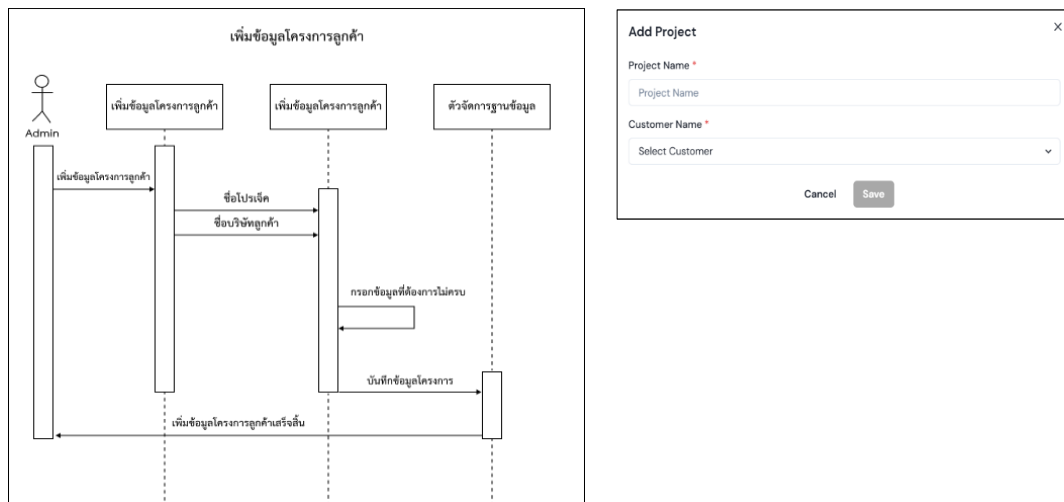
ภาพที่ 1 แสดงฟังก์ชันการทำงานของระบบ

1) การจัดการข้อมูลลูกค้า (Customer Management Module) ผู้ดูแลระบบมีหน้าที่เพิ่มข้อมูลบริษัทลูกค้า โดยเข้าสู่หน้า "ลูกค้า" ในระบบ ซึ่งจะแสดงฟอร์มสำหรับกรอกข้อมูลบริษัท เมื่อกรอกข้อมูลครบถ้วนและบันทึก ระบบจะเก็บข้อมูลบริษัทไว้เพื่อใช้ในขั้นตอนต่อไป แสดงดังภาพที่ 2



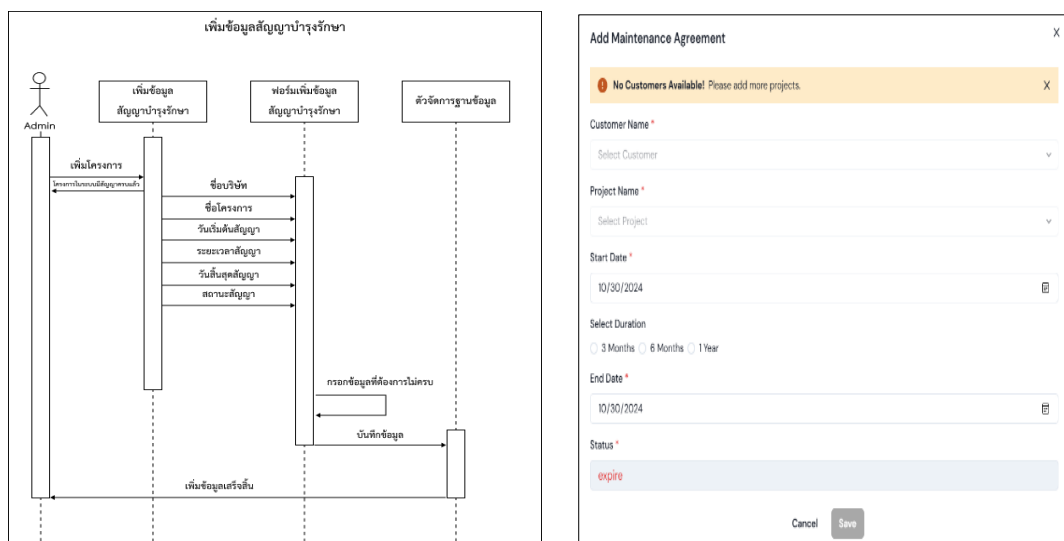
ภาพที่ 2 ขั้นตอนและหน้าจอการเพิ่มข้อมูลลูกค้า

2) การจัดการโครงการ (Project Management) เมื่อมีข้อมูลบริษัทลูกค้าในระบบแล้ว ผู้ดูแลระบบสามารถเพิ่มข้อมูลโครงการได้ โดยเข้าสู่ฟอร์มการเพิ่มข้อมูลโครงการและกรอกข้อมูลที่เกี่ยวข้อง เช่น ชื่อโครงการ รายละเอียด และข้อมูลอื่น ๆ ที่จำเป็น แสดงดังภาพที่ 3



ภาพที่ 3 ขั้นตอนและหน้าจอการเพิ่มข้อมูลโครงการลูกค้า

3) การจัดการสัญญาบำรุงรักษา (Contract Management) เมื่อลูกค้ามีการทำสัญญาบำรุงรักษา ผู้ดูแลระบบสามารถกรอกข้อมูลสัญญาบำรุงรักษาได้ โดยระบุรายละเอียด เช่น ระยะเวลาการทำสัญญา และเงื่อนไขต่าง ๆ ผ่านฟอร์มในระบบแสดงดังภาพที่ 4



ภาพที่ 4 ขั้นตอนและหน้าจอการเพิ่มข้อมูลสัญญาบำรุงรักษา

4) การจัดการต่ออายุสัญญา (Renewal Contract Management) การต่ออายุสัญญาเมื่อสถานะของสัญญาบำรุงรักษาใกล้หมดอายุ จะมีการส่งแจ้งเตือนเป็นระยะเวลาภายใน 30 วัน 15 วัน และ 7 วัน นับจากวันปัจจุบันจนถึงวันหมดอายุสัญญา ซึ่งถ้าหากผู้ดูแลระบบบรรเทาและมีการติดต่อลูกค้าแล้วจะมีการตกลงการต่ออายุสัญญาบำรุงรักษา ทางผู้ดูแลระบบจะสามารถต่อสัญญาได้ โดย การเข้าหน้าส่วนการดูแลลูกค้า เลือกส่วนหน้าต่างสัญญาบำรุงรักษาและกรอกฟอร์มต่อสัญญาบำรุงรักษา แสดงดังภาพที่ 5

ภาพที่ 5 หน้าแสดงผลการเพิ่มบัญชีผู้ใช้งานและการจัดการต่ออายุสัญญา

5) การจัดการคำถามและปัญหา (Enquiry & Request Handling) การบันทึกคำถามทำได้โดยเข้าหน้าส่วนการดูแลโครงการ แล้วกดปุ่มเพื่อกรอกฟอร์มบันทึกปัญหาเลือกประเภทเป็นคำถาม หากต้องการบันทึกปัญหาเพื่อส่งมอบให้ทีมพัฒนาระบบเป็นผู้แก้ไขปัญหาให้เลือกประเภทเป็นปัญหา แล้วกดปุ่มไปที่หน้าการแก้ไขปัญหาเพื่ออัปเดตการแก้ปัญหาในหน้าต่างใหม่ ซึ่งในส่วนของหน้าการแก้ไขปัญหา สามารถกรอกการแก้ไขปัญหา สถานการณ์แก้ไขรวมถึงรายละเอียดรูปภาพต่างๆด้วย แสดงดังภาพที่ 6

ภาพที่ 6 หน้าแสดงผลการจัดการปัญหาและคำถาม

5.2 ชั้นตรรกะของระบบ (Application Layer) การคำนวณวันหมดอายุของสัญญา การส่งการแจ้งเตือนในหลายระยะเวลา เช่น 30, 15, และ 7 วันก่อนครบอายุ ระบบสามารถตั้ง Cron job หรือ Scheduled Task ให้ทำงานทุกวันเพื่อตรวจสอบ days_left และส่งการแจ้งเตือนตามช่วงเวลา 30, 15, และ 7 วัน ซึ่งสอดคล้องกับ Notification Theory แบบ Multi-phase reminders ที่เน้นการแจ้งเตือนซ้ำในช่วงเวลาที่เหมาะสม เพื่อเพิ่มโอกาสในการต่อสัญญา แสดงอัลกอริทึมดังนี้

```
const moment = require('moment');
contracts.forEach(contract => {
  const expiryDate = moment(contract.startDate).add(contract.durationMonths,
'months');
  const today = moment();
  const daysLeft = expiryDate.diff(today, 'days');
  if ([30, 15, 7, 0].includes(daysLeft)) {
    notifyUser(contract.userId, contract.id, `สัญญาจะหมดอายุในอีก ${daysLeft}
วัน`);} });
```

5.3 ชั้นฐานข้อมูล (Data Layer) ระบบการจัดการสัญญาบำรุงรักษาที่ออกแบบใหม่นี้เน้นการทำงานอัตโนมัติในการแจ้งเตือนและต่ออายุสัญญา โดยมีโครงสร้างหลักที่เชื่อมโยงกันผ่าน Entity-Relationship Diagram แบบ Crow's Foot เริ่มจาก Customer ซึ่งเป็นศูนย์กลางของระบบ สามารถมีได้หลาย Project (ความสัมพันธ์แบบหนึ่งต่อหลาย) แต่ละ Project ก็สามารถมีได้หลาย Contract เช่นเดียวกัน เมื่อ Contract ใกล้ถึงวันหมดอายุ (End Date) ระบบจะสร้าง Renewal Notification อัตโนมัติในสามระยะเวลาดังกล่าว ได้แก่ 30 วัน, 15 วัน และ 7 วันก่อนหมดอายุ โดยแต่ละ Notification จะบันทึกประเภทการแจ้งเตือน (Notification Type) และวันที่ส่ง (Sent Date) ส่วนสำคัญคือการต่ออายุสัญญา (Renewal) ที่จะถูกสร้างขึ้นเมื่อลูกค้ายืนยันการต่ออายุ โดย Renewal จะเชื่อมโยงกลับไป Notification ที่เกี่ยวข้อง ทำให้สามารถติดตามได้ว่าเกิดจากการแจ้งเตือนครั้งใด นอกจากนี้ระบบยังมี Entities อื่นๆ ที่สำคัญ เช่น Maintenance Service สำหรับบันทึกรายการบริการบำรุงรักษา, Enquiry และ Response สำหรับจัดการคำถามและปัญหาของลูกค้า รวมถึง Contact Person สำหรับเก็บข้อมูลผู้ติดต่อของลูกค้าแต่ละราย การออกแบบนี้ทำให้ระบบสามารถจัดการวงจรชีวิตของสัญญาได้อย่างครบถ้วน ตั้งแต่การแจ้งเตือนล่วงหน้า การต่ออายุสัญญา ไปจนถึงการบันทึกประวัติการดำเนินงานทั้งหมด ซึ่งช่วยเพิ่มประสิทธิภาพการทำงานและลดโอกาสผิดพลาดจากการทำงานด้วยมนุษย์ แสดงดังภาพที่ 7



ภาพที่ 7 แสดง Entity-Relationship Diagram

6. ผลการวิจัย

6.1 สัญญาที่ครบกำหนดในแต่ละเดือน จากตารางที่ 1 ซึ่งแสดงจำนวนสัญญาที่ครบกำหนดในแต่ละเดือนของปี 2567 แบ่งตามบริษัทและประเภทสัญญา (3 เดือน, 6 เดือน, 12 เดือน) พบว่า 1) ภาพรวมสัญญาที่ครบกำหนด จำนวนสัญญาทั้งหมด มีสัญญาครบกำหนดทั้งสิ้น 17 สัญญา ตลอดปี 2567 ประเภทสัญญาประกอบด้วย สัญญา 3 เดือน จำนวน 6 สัญญา (35.3%) สัญญา 6 เดือน จำนวน 8 สัญญา (47.1%) และ สัญญา 12 เดือน จำนวน 3 สัญญา (17.6%) โดยสัญญา 6 เดือน มีจำนวนมากที่สุด แสดงถึง ความนิยมหรือความเหมาะสมของระยะเวลานี้สำหรับการทำสัญญาในกลุ่มบริษัทเหล่านี้ 2) การกระจายตัวของสัญญาตามเดือน เดือนที่มีสัญญาครบกำหนดมากที่สุด ประกอบด้วยเดือน 3, 5, 6, และ 7 มีสัญญาครบกำหนดเดือนละ 2 สัญญา เดือน 1, 4, 8, 10, และ 12 มีสัญญาครบกำหนดเดือนละ 1-2 สัญญา ส่วนเดือน 11 ไม่มีสัญญาครบกำหนด ข้อสังเกตคือ การกระจายตัวของสัญญาค่อนข้างสม่ำเสมอในช่วงครึ่งแรกของปี (ม.ค.-ก.ค.) โดยมีสัญญาครบกำหนดทุกเดือน ครึ่งหลังของปี (ส.ค.-ธ.ค.) มีสัญญาครบกำหนดน้อยลง โดยเฉพาะเดือน 11 ที่ไม่มีสัญญาเลย เดือน 3 และ 6 มีจำนวนสัญญาครบกำหนดสูง อาจบ่งชี้ถึงการเริ่มต้นสัญญาในช่วงปลายปี 2566 หรือ ต้นปี 2567

ตารางที่ 1 แสดงสัญญาที่ครบกำหนดในแต่ละเดือน ปี 2567

บริษัท	เดือน (สัญญา 3 เดือน 6 เดือน และ 12 เดือน)												รวม
	1	2	3	4	5	6	7	8	9	10	11	12	
1	-	-	3	-	-	6	-	-	3	-	-	-	3
2	-	6	-	-	6	-	-	12	-	-	-	-	3
3	3	-	-	3	6	-	6	-	-	-	-	-	4
4	-	-	12	-	-	6	-	-	-	-	-	-	2
5	6	-	-	3	-	-	6	-	-	3	-	12	5
ระยะเวลา	1	2	3	4	5	6	7	8	9	10	11	12	รวม
3 M.	1	0	1	2	0	0	0	0	1	1	0	0	6
6 M.	1	1	0	0	2	2	2	0	0	0	0	0	8
12 M.	0	0	1	0	0	0	0	1	0	0	0	1	3
รวม	2	1	2	2	2	2	2	1	1	1	0	1	17

6.2 ข้อมูลการแจ้งเดือนสัญญาที่ครบกำหนดภายใน 30 วัน และการต่อสัญญา จากตารางที่ 2 ซึ่งแสดงข้อมูลการแจ้งเดือนสัญญาที่ครบกำหนดในปี 2567 และการต่อสัญญาของแต่ละบริษัท ภาพรวมสัญญาที่ครบกำหนดและการต่อสัญญา จำนวนสัญญาที่ครบกำหนดทั้งหมด 17 สัญญา จำนวนสัญญาที่ต่อสัญญาได้ จำนวน 7 สัญญา อัตราการต่อสัญญาโดยรวม 45.67% ข้อสังเกตคือ อัตราการต่อสัญญาค่อนข้างต่ำ (น้อยกว่าครึ่งหนึ่งของสัญญาทั้งหมด) ซึ่งอาจบ่งชี้ถึงความท้าทายในการรักษาลูกค้าหรือการตัดสินใจไม่ต่อสัญญาเนื่องจากปัจจัยต่าง ๆ เช่น ความพึงพอใจ ราคา หรือความต้องการที่เปลี่ยนไป **การวิเคราะห์ตามบริษัท** บริษัท 1 สัญญาครบกำหนด จำนวน 3 สัญญา (3 เดือน 2 สัญญา, 6 เดือน 1 สัญญา) การต่อสัญญา จำนวน 3 สัญญา (100%) โดยบริษัท 1 มีอัตราการต่อสัญญาสูงสุด (100%) แสดงถึงความสำเร็จในการรักษาความสัมพันธ์กับลูกค้าหรือความพึงพอใจในสัญญา บริษัท 2 สัญญาครบกำหนด จำนวน 3 สัญญา (6 เดือน 2 สัญญา, 12 เดือน 1 สัญญา) การต่อสัญญา จำนวน 1 สัญญา (33.33%) โดยอัตราการต่อสัญญานต่ำ อาจบ่งชี้ถึงปัญหาในสัญญา 6 เดือนที่ไม่ได้รับการต่ออายุ บริษัท 3 สัญญาครบกำหนด จำนวน 4 สัญญา (3 เดือน 2 สัญญา, 6 เดือน 2 สัญญา) การต่อสัญญา จำนวน 1 สัญญา (25%) โดยอัตราการต่อสัญญานต่ำมาก อาจสะท้อนถึงความไม่พึงพอใจหรือการเปลี่ยนแปลงความต้องการของลูกค้า บริษัท 4 สัญญาครบกำหนด จำนวน 2 สัญญา (6 เดือน 1 สัญญา, 12 เดือน 1 สัญญา) การต่อสัญญา จำนวน 1 สัญญา (50%) โดยอัตราการต่อสัญญานปานกลาง สัญญา 12 เดือนได้รับการต่อ แต่สัญญา 6 เดือนไม่ได้รับ

การต่อ บริษัท 5 สัญญาครบกำหนด จำนวน 5 สัญญา (3 เดือน 2 สัญญา, 6 เดือน 2 สัญญา, 12 เดือน 1 สัญญา) การต่อสัญญา จำนวน 1 สัญญา (20%) โดยอัตราการต่อสัญญาต่ำที่สุด แม้จะมีสัญญาครบกำหนดมากที่สุด อาจบ่งชี้ถึงปัญหาในการบริหารสัญญาหรือความสัมพันธ์กับลูกค้า

ตารางที่ 2 แสดงการแจ้งเตือนสัญญาที่ครบกำหนด ภายใน 30 วัน (แจ้งเตือนระยะที่ 1)

บริษัท	สัญญาที่ครบอายุ	ต่อสัญญา	เตือนครั้งที่ 1	ได้ต่อสัญญา	ร้อยละ
1	3(1), 6(1), 3(1)	3(v), 6(v), 3(v)	3	3	100.00
2	6(1), 6(1), 12(1)	6(x), 6(x), 12(v)	3	1	33.33
3	3(1), 3(1), 6(1), 6(1)	3(x), 3(x), 6(v), 6(x)	4	1	25.00
4	6(1), 12(1)	6(x), 12(v)	2	1	50.00
5	6(1), 3(1), 6(1), 3(1), 12(1)	6(x), 3(x), 6(v), 3(x), 12(x)	5	1	20.00
รวม (สัญญา)			17	7	45.67

หมายเหตุ: v หมายถึง ได้ต่อสัญญา x หมายถึง ไม่ได้ต่อสัญญา

$$\text{เปอร์เซ็นต์ความสำเร็จ} = \left(\frac{\text{ต่อสัญญาสำเร็จ}}{\text{สัญญาที่ครบอายุคงเหลือทั้งหมด แจ้งเตือนภายใน 30 วัน}} \right) \times 100 \quad 1$$

การวิเคราะห์ตามประเภทสัญญา สัญญา 3 เดือน (6 สัญญา) การต่อสัญญา จำนวน 3 สัญญา (50%) ข้อสังเกตคือ สัญญา 3 เดือนมีอัตราการต่อสัญญาค่อนข้างดี โดยเฉพาะในบริษัท 1 ที่ต่อสัญญาทั้งหมด สัญญา 6 เดือน (8 สัญญา) การต่อสัญญา จำนวน 2 สัญญา (25%) ข้อสังเกตคือ สัญญา 6 เดือนมีอัตราการต่อสัญญาต่ำ แม้ว่าจะเป็นประเภทสัญญาที่มีจำนวนมากที่สุด อาจบ่งชี้ถึงปัญหาในระยะเวลาหรือเงื่อนไขของสัญญานี้ สัญญา 12 เดือน (3 สัญญา) การต่อสัญญา: 2 สัญญา (66.67%) ข้อสังเกตคือ สัญญา 12 เดือนมีอัตราการต่อสัญญาสูงสุด อาจสะท้อนถึงความมั่นคงหรือความพึงพอใจในสัญญาระยะยาว **ประสิทธิภาพของการแจ้งเตือน** การแจ้งเตือนสัญญาครบกำหนดภายใน 30 วันมีจำนวนเท่ากับสัญญาที่ครบกำหนด (17 ครั้ง) แสดงว่าระบบการแจ้งเตือนทำงานครบถ้วน อย่างไรก็ตาม การแจ้งเตือนไม่ได้นำไปสู่การต่อสัญญาในทุกกรณี โดยเฉพาะในบริษัท 3 และ 5 ที่มีอัตราการต่อสัญญาต่ำมาก โดยการแจ้งเตือนอาจไม่เพียงพอต่อการกระตุ้นให้ลูกค้าต่อสัญญา อาจต้องพิจารณากลยุทธ์เพิ่มเติม เช่น การติดตามผลเชิงรุกหรือการนำเสนอข้อเสนอพิเศษ

6.3 ข้อมูลการแจ้งเตือนสัญญาที่ครบกำหนดภายใน 15 วัน และการต่อสัญญา จากตารางที่ 3 ซึ่งแสดงข้อมูลการแจ้งเตือนสัญญาที่ครบกำหนดในปี 2567 โดยเป็นการแจ้งเตือนครั้งที่ 2

ภายใน 15 วันหลังจากแจ้งเตือนครั้งแรก ภาพรวมสัญญาที่ได้รับการแจ้งเตือนครั้งที่ 2 และการต่อสัญญา จำนวนสัญญาที่ได้รับการแจ้งเตือนครั้งที่ 2 คงเหลือจำนวน 10 สัญญา จำนวนสัญญาที่ต่อสัญญาได้ จำนวน 4 สัญญา อัตราการต่อสัญญาโดยรวม: 52.08% โดยอัตราการต่อสัญญาหลังการแจ้งเตือนครั้งที่ 2 (52.08%) สูงกว่าอัตราการต่อสัญญาหลังการแจ้งเตือนครั้งแรก (45.67% จากตารางที่ 5 ก่อนหน้า) แสดงว่าการแจ้งเตือนครั้งที่ 2 ช่วยกระตุ้นให้ลูกค้าบางรายตัดสินใจต่อสัญญา อย่างไรก็ตาม ยังมีสัญญาครึ่งหนึ่งที่ไม่ได้รับการต่ออายุ แม้จะมีการแจ้งเตือนซ้ำ

ตารางที่ 3 แสดงการแจ้งเตือนสัญญาที่ครบอายุ ภายใน 15 วัน (แจ้งเตือนระยะที่ 2)

บริษัท	สัญญาที่ครบอายุ	ต่อสัญญา	เตือนครั้งที่ 2	ต่อสัญญา	ร้อยละ
2	6(2), 6(2)	6(x), 6(v)	2	1	50.00
3	3(2), 3(2), 6(2)	3(x), 3(v), 6(x)	3	1	33.33
4	6(2)	6(v)	1	1	100.00
5	6(2), 3(2), 3(2), 2(2)	6(x), 3(x), 3(v), 12(x)	4	1	25.00
รวม (สัญญา)			10	4	52.08

หมายเหตุ: v หมายถึง ได้ต่อสัญญา x หมายถึง ไม่ได้ต่อสัญญา

$$\text{ร้อยละความสำเร็จ} = \left(\frac{\text{ต่อสัญญาสำเร็จ}}{\text{สัญญาที่ครบอายุคงเหลือทั้งหมด แจ้งเตือนภายใน 15 วัน}} \right) \times 100 \quad 2$$

การวิเคราะห์ตามบริษัท บริษัท 1 สัญญาครบกำหนด ดังนั้นจึงไม่มีการแจ้งเตือนครั้งที่ 2 บริษัท 2 สัญญาครบกำหนด จำนวน 2 สัญญา (6 เดือน 2 สัญญา) ต่อสัญญา จำนวน 1 สัญญา (50%) โดยอัตราการต่อสัญญาเพิ่มขึ้นจาก 33.33% (หลังแจ้งเตือนครั้งแรก) เป็น 50% หลังแจ้งเตือนครั้งที่ 2 แสดงว่าการแจ้งเตือนครั้งที่ 2 มีผลในระดับหนึ่ง บริษัท 3 สัญญาครบกำหนด จำนวน 3 สัญญา (3 เดือน 2 สัญญา, 6 เดือน 1 สัญญา) ต่อสัญญา จำนวน 1 สัญญา (33.33%) โดยอัตราการต่อสัญญาเพิ่มจาก 25% (หลังแจ้งเตือนครั้งแรก) เป็น 33.33% การแจ้งเตือนครั้งที่ 2 ช่วยให้ต่อสัญญาเพิ่ม 1 สัญญา แต่โดยรวมยังต่ำ บริษัท 4 สัญญาครบกำหนด จำนวน 1 สัญญา (6 เดือน 1 สัญญา) ต่อสัญญา จำนวน 1 สัญญา (100%) โดยบริษัท 4 มีประสิทธิภาพสูงในการต่อสัญญาหลังแจ้งเตือนครั้งที่ 2 (100%) สอดคล้องกับอัตราการต่อสัญญา 50% หลังแจ้งเตือนครั้งแรก บริษัท 5 สัญญาครบกำหนด จำนวน 4 สัญญา (3 เดือน 2 สัญญา, 6 เดือน 1 สัญญา, 12 เดือน 1 สัญญา) ต่อสัญญา จำนวน 1 สัญญา (25%) โดยอัตราการต่อสัญญายังคงต่ำ (เพิ่มจาก 20% เป็น 25%) แม้จะมีการแจ้งเตือนครั้งที่ 2 และมีสัญญาครบกำหนดมากที่สุด แสดงถึงความท้าทายในการรักษาลูกค้า **การวิเคราะห์ตามประเภทสัญญา** สัญญา 3 เดือน (4 สัญญา) ต่อสัญญา จำนวน 2 สัญญา (50%) โดย

สัญญา 3 เดือนมีอัตราการต่อสัญญาค่อนข้างดีหลังแจ้งเตือนครั้งที่ 2 เทียบเท่ากับอัตราการต่อสัญญา หลังแจ้งเตือนครั้งแรก (50% จากตารางก่อนหน้า) สัญญา 6 เดือน (5 สัญญา) ต่อสัญญา จำนวน 2 สัญญา (40%) โดยอัตราการต่อสัญญาเพิ่มขึ้นจาก 25% (หลังแจ้งเตือนครั้งแรก) เป็น 40% การแจ้งเตือนครั้งที่ 2 ช่วยกระตุ้นการต่อสัญญาในสัญญา 6 เดือน สัญญา 12 เดือน (1 สัญญา) ต่อสัญญา จำนวน 0 สัญญา (0%) โดยสัญญา 12 เดือนไม่ได้รับการต่อสัญญาเลยหลังแจ้งเตือนครั้งที่ 2 แม้ว่า ตารางก่อนหน้าจะแสดงอัตราการต่อสัญญา 66.67% (2 จาก 3 สัญญา) อาจบ่งชี้ถึงปัญหาเฉพาะใน สัญหานี้

6.4 ข้อมูลการแจ้งเตือนสัญญาที่ครบกำหนดภายใน 7 วัน และการต่อสัญญา ตารางที่ 4 แสดงข้อมูลการแจ้งเตือนสัญญาครบกำหนดครั้งที่ 3 ภายใน 7 วันในปี 2567 ภาพรวมสัญญาที่ได้รับ การแจ้งเตือนครั้งที่ 3 และการต่อสัญญาจำนวนสัญญาที่ได้รับการแจ้งเตือนครั้งที่ 3 เหลือจำนวน 6 สัญญา จำนวนสัญญาที่ต่อสัญญา จำนวน 2 สัญญา อัตราการต่อสัญญาโดยรวม 22.22% (จากตาราง) โดยอัตราการต่อสัญญาหลังการแจ้งเตือนครั้งที่ 3 (22.22%) ลดลงอย่างมากเมื่อเทียบกับการแจ้งเตือนครั้งที่ 2 (52.08%) และครั้งแรก (45.67%) แสดงว่าการแจ้งเตือนครั้งที่ 3 มีประสิทธิภาพต่ำในการกระตุ้นการต่อสัญญา จำนวนสัญญาที่ต้องแจ้งเตือนครั้งที่ 3 ลดลงเหลือ 6 สัญญา (จาก 10 สัญญาในการแจ้งเตือนครั้งที่ 2) ซึ่งสอดคล้องกับสัญญาที่ไม่ต่อหลังแจ้งเตือนครั้งที่ 2 **การวิเคราะห์ตามบริษัท** บริษัท 1 ต่อสัญญาครบทั้งหมดในการแจ้งเตือนครั้งที่ 1 (100%) จึงไม่มีการแจ้งเตือนครั้งที่ 3 บริษัท 2 สัญญาครบกำหนด จำนวน 1 สัญญา (6 เดือน 1 สัญญา) ต่อสัญญา จำนวน 0 สัญญา (0%) โดยไม่มีการต่อสัญญาเลยหลังแจ้งเตือนครั้งที่ 3 (ลดจาก 50% ในการแจ้งเตือนครั้งที่ 2) แสดงว่าสัญญานี้อาจมีปัญหาที่ทำให้ลูกค้าไม่ต่อสัญญา บริษัท 3 สัญญาครบกำหนด จำนวน 2 สัญญา (3 เดือน 1 สัญญา, 6 เดือน 1 สัญญา) ต่อสัญญา จำนวน 0 สัญญา (0%) โดยอัตราการต่อสัญญาลดลง จาก 33.33% (ครั้งที่ 2) เป็น 0% แสดงว่าการแจ้งเตือนครั้งที่ 3 ไม่มีผลต่อการตัดสินใจของลูกค้า บริษัท 4 ต่อสัญญาครบทั้งหมดในการแจ้งเตือนครั้งที่ 2 (100%) จึงไม่มีการแจ้งเตือนครั้งที่ 3 บริษัท 5 สัญญาครบกำหนด จำนวน 3 สัญญา (3 เดือน 1 สัญญา, 6 เดือน 1 สัญญา, 12 เดือน 1 สัญญา) ต่อสัญญา จำนวน 2 สัญญา (66.67%) โดยบริษัท 5 มีอัตราการต่อสัญญาสูงสุดในการแจ้งเตือนครั้งที่ 3 (เพิ่มจาก 25% ในครั้งที่ 2) โดยเฉพาะสัญญา 3 เดือนและ 12 เดือน แสดงว่าการแจ้งเตือนครั้งที่ 3 มีผลต่อลูกค้าบางราย

ตารางที่ 4 แสดงการแจ้งเตือนสัญญาที่ครบอายุ ภายใน 7 วัน (แจ้งเตือนระยะที่ 3)

บริษัท	สัญญาที่ครบอายุ	ต่อสัญญา	เตือนครั้งที่ 3	ต่อสัญญา	ร้อยละ
2	6(3)	6(x)	1	0	0.00
3	3(3), 6(3)	3(x), 6(x)	2	0	0.00
5	6(3), 3(3), 12(3)	6(x), 3(v), 12(v)	3	2	66.67
รวม (สัญญา)			6	2	22.22

หมายเหตุ: √ หมายถึง ได้ต่อสัญญา × หมายถึง ไม่ได้ต่อสัญญา

$$\text{ร้อยละความสำเร็จ} = \left(\frac{\text{ต่อสัญญาสำเร็จ}}{\text{สัญญาที่ครบอายุคงเหลือทั้งหมด แจ้งเตือนภายใน 7 วัน}} \right) \times 100 \quad 3$$

6.5 การเปรียบเทียบประสิทธิภาพระหว่างบริษัท จากตารางที่ 5 แสดงการเปรียบเทียบประสิทธิภาพระหว่างบริษัท บริษัทที่มีประสิทธิภาพสูงสุดประกอบด้วย บริษัท 1 (100.00%) ต่อสัญญาครบทุกสัญญาด้วยการแจ้งเตือนเพียงครั้งเดียว และบริษัท 4 (66.67%) ต่อสัญญาได้เกือบทั้งหมดด้วยจำนวนการแจ้งเตือนที่น้อย ส่วนบริษัทที่มีประสิทธิภาพต่ำประกอบด้วย บริษัท 3 (22.22%) มีการแจ้งเตือนมากแต่ต่อสัญญาได้น้อย และบริษัท 2 และ 5 (33.33%) มีประสิทธิภาพปานกลาง แต่ต้องใช้การแจ้งเตือนจำนวนมาก โดยที่บริษัท 1 และ 4 ใช้จำนวนการแจ้งเตือนน้อย (3 ครั้ง) แต่ได้ผลลัพธ์ดี แสดงถึงกลยุทธ์การบริหารสัญญาที่มีประสิทธิภาพ ส่วนบริษัท 3 และ 5 ใช้การแจ้งเตือนมาก (9 และ 12 ครั้ง) แต่ผลลัพธ์ต่ำ แสดงถึงปัญหาในการโน้มน้าวลูกค้าหรือเงื่อนไขสัญญาที่ไม่เหมาะสม

ตารางที่ 5 การเปรียบเทียบประสิทธิภาพระหว่างบริษัท

บริษัท	การแจ้งเตือน (ครั้ง)	ต่อสัญญาสำเร็จ (สัญญา)	ร้อยละความสำเร็จ
1	3	3	100.00
2	6	2	33.33
3	9	2	22.22
4	3	2	66.67
5	12	4	33.33
รวม	33	13	39.39

$$\text{ร้อยละความสำเร็จ} = \left(\frac{\text{ต่อสัญญาสำเร็จ}}{\text{การแจ้งเตือน}} \right) \times 100$$

4

6.6 ความสำเร็จสะสมของการต่อสัญญา ตารางที่ 6 แสดงภาพรวมความสำเร็จสะสมจำนวนสัญญาครบกำหนดทั้งหมด จำนวน 17 สัญญา (จากตารางที่ 4) จำนวนสัญญาที่ต่อสัญญาสำเร็จทั้งหมด จำนวน 13 สัญญา เปอร์เซ็นต์ความสำเร็จสะสมสูงสุด 76.47% (หลังการแจ้งเตือนครั้งที่ 3) อัตราการต่อสัญญาเพิ่มขึ้นตามจำนวนครั้งของการแจ้งเตือน จาก 41.18% (ครั้งที่ 1) เป็น 64.71% (ครั้งที่ 2) และ 76.47% (ครั้งที่ 3) อย่างไรก็ตาม ยังมี 4 สัญญา (23.53%) ที่ไม่ได้รับการต่อสัญญาแม้จะมีการแจ้งเตือนถึง 3 ครั้ง แสดงถึงข้อจำกัดของระบบการแจ้งเตือนในการโน้มน้าวลูกค้าทั้งหมด แม้จะมีการแจ้งเตือนถึง 3 ครั้ง แต่ยังมี 4 สัญญา (23.53%) ที่ไม่ต่อสัญญา สะท้อนถึงปัญหาที่อาจอยู่นอกเหนือจากการแจ้งเตือน เช่น เงื่อนไขสัญญา คุณภาพบริการ หรือความต้องการของลูกค้าที่เปลี่ยนไป การแจ้งเตือนครั้งที่ 3 มีประสิทธิภาพต่ำ (2 จาก 6 สัญญา) ซึ่งอาจบ่งชี้ว่าลูกค้าที่ไม่ต่อสัญญาหลังครั้งที่ 2 มักตัดสินใจไม่ต่อสัญญาแล้ว

7. บทสรุปและอภิปรายผลการวิจัย

ผลการแจ้งเตือนทั้งสามระยะ: การแจ้งเตือนสามระยะ (เตือน 1 ครั้งใน 30 วัน, 2 ครั้งใน 15 วัน, และ 3 ครั้งใน 7 วัน) มีสัญญาครบกำหนดรวม 33 สัญญา ต่อสัญญาสำเร็จ 13 สัญญา คิดเป็นอัตราความสำเร็จเฉลี่ย 39.39% เตือนครั้งที่ 1 ต่อสัญญา 7 จาก 17 สัญญา (41.18%), เตือนครั้งที่ 2 ต่อสัญญา 4 จาก 10 สัญญา (40%), และเตือนครั้งที่ 3 ต่อสัญญา 2 จาก 6 สัญญา (33.33%) ความสำเร็จสะสมสูงสุดที่ 76.47% หลังเตือน 3 ครั้ง โดยการเตือน 2 ครั้งมีประสิทธิภาพต่อรอบสูงสุด ส่วนการเตือน 3 ครั้งมีประสิทธิภาพต่ำสุด อาจเนื่องจากระยะเวลาสั้นเกินไป ประสิทธิภาพของบริษัทและความสำเร็จสะสม: บริษัท 1 มีประสิทธิภาพสูงสุด (100%) ต่อสัญญาครบทั้ง 3 สัญญา รองลงมาคือบริษัท 4 (66.67%) ต่อสัญญา 2 จาก 3 ครั้ง บริษัท 2 และ 5 มีอัตราความสำเร็จปานกลาง (33.33%) ส่วนบริษัท 3 ต่ำสุด (22.22%) การแจ้งเตือนหลายรอบช่วยเพิ่มโอกาสต่อสัญญา แต่ 23.53% ของสัญญายังไม่ต่อ ควรปรับปรุงกลยุทธ์การแจ้งเตือนและเงื่อนไขสัญญาในบริษัท 3 และ 5 เพื่อเพิ่มประสิทธิภาพ การวิเคราะห์การแจ้งเตือนสัญญาทั้งสามระยะและประสิทธิภาพบริษัท มีความสอดคล้องงานวิจัยของ ณพทัยวรรณ วิโสภา และคณะ (2564) ที่ทำการพัฒนาระบบจัดการข้อมูลที่มีการแจ้งเตือนล่วงหน้า ช่วยลดความล่าช้าในการต่อสัญญา คล้ายกับระบบในงานนี้ที่ต่อสัญญาสำเร็จ 13 จาก 33 สัญญา (39.39%) งานวิจัยของ Kumar et al. (2024) และ Brown et al. (2023) สนับสนุนการใช้ Agile และ feedback loop ซึ่งคล้ายกับการเตือนหลายรอบที่เพิ่มความสำเร็จสะสมถึง 76.47% Wang et al. (2024) และ Saito et al. (2024) เสนอ automation และ predictive

analytics เพื่อเพิ่มประสิทธิภาพ คล้ายกับการจัดการสัญญาของบริษัท 1 (100%) และ 4 (66.67%)
 คำนวณแนวทางเหล่านี้ไปปรับใช้ในบริษัท 3 และ 5 เพื่อลดสัญญาที่ไม่ต่อ 23.53%

ตารางที่ 6 การคำนวณความสำเร็จการต่อสัญญาสะสมตามระยะเวลาการแจ้งเตือน

ระยะ (วัน)	สัญญาครบอายุ	ต่อสัญญาสำเร็จ	สำเร็จสะสม	ร้อยละความสำเร็จสะสม
30	17.00	7.00	7.00	41.18
15	10.00	4.00	11.00	64.71
7	6.00	2.00	13.00	76.47

$$\text{ร้อยละความสำเร็จสะสม} = \left(\frac{\text{ต่อสัญญาสำเร็จสะสม}}{\text{สัญญาที่ครบอายุทั้งหมด}} \right) \times 100 \quad 5$$

8. กิตติกรรมประกาศ

ข้าพเจ้าขอแสดงความขอบคุณอย่างยิ่งต่อ บริษัท ลูคัส สเตรททิจี จำกัด ที่ได้ให้การสนับสนุนความช่วยเหลือด้านทรัพยากรและความรู้ในการดำเนินการวิจัย การสนับสนุนอันมีค่านี้มีส่วนสำคัญอย่างยิ่งต่อความสำเร็จและเป็นที่ประจักษ์ถึงความมุ่งมั่นของบริษัท

9. เอกสารอ้างอิง

- ณทัยวรรณ วิโสภา, สุธิตเทพ ศิริพิพัฒนกุล, และ ณัฐพล รำไพ. (2564). การพัฒนาระบบฐานข้อมูล
 เพื่อการจัดการพัสดุภาควิชาเทคโนโลยีการศึกษา คณะศึกษาศาสตร์ มหาวิทยาลัย
 เกษตรศาสตร์. **วารสารเทคโนโลยีสื่อสารมวลชน มทร.พระนคร**, 6(2), 63–74.
- สมนึก สีนุปวน, ชวัลวิทย์ เลาหาง, ณัฐชยา เตปินวงศ์, พุฒิพงศ์ พนมไพร, ก่องกาญจน์ ดุลยไชย,
 และอรรณวิธ ชังคมานนท์. (2568). ระบบรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและ
 การแจ้งเตือนผ่านไลน์. **วารสารแม่โจ้เทคโนโลยีสารสนเทศและนวัตกรรม**, 11(1), 83–100.
- อรรถพล จันทร์สมุด. (2563). ระบบโซ่อุปทานแบบดิจิทัลเพื่อการจัดการโครงการใน
 สถาบันอุดมศึกษา. **วารสารแม่โจ้เทคโนโลยีสารสนเทศและนวัตกรรม**, 6(1), 1–9.
- Ahmed, S., Torres, M., & Choi, E. (2022). Soft computing techniques for software
 maintainability prediction: A hybrid approach. **IEEE Transactions on Software
 Engineering**, 48(3), 1023-1045.
- Brown, T., Nguyen, H., & Alvarez, L. (2023). Integrated software maintenance

- framework for agile environments. **Journal of Systems and Software**, 195, 111501.
- Chen, X., Rahman, M., & Silva, P. (2024). Quality control in outsourced software maintenance: An empirical study. **Information and Software Technology**, 158, 107203.
- Fernández, D., Kim, H., & Singh, V. (2024). Hot patching techniques for zero-downtime software maintenance. **IEEE Access**, 12, 12345-12360.
- Kumar, A., Sharma, R., & Lee, J. (2024). Agile practices for cloud-based software maintenance: A systematic review. **Journal of Cloud Computing**, 13(1), 1-22.
- Lim, C., Fernández, J., & Wang, Q. (2023). Cost-benefit prioritization model for SME software maintenance. **Software: Practice and Experience**, 53(8), 1450-1468.
- Rossi, F., Nakamura, K., & Wright, D. (2024). Cost optimization model for software maintenance budgeting. **Journal of Software Maintenance and Evolution**, 36(2), e2478.
- Saito, Y., Gupta, S., & Al-Mutairi, N. (2024). Reliability-centered maintenance for IoT software systems. **Future Generation Computer Systems**, 151, 345-360.
- Wang, L., Zhang, Y., & Patel, M. (2024). Maintenance overhead of GitHub Actions workflows: An empirical study. **Empirical Software Engineering**, 29(1), 1-30.