Received: 3 พ.ค. 2567 **Revised:** 13 ส.ค. 2567 **Accepted:** 16 ส.ค. 2567

ระบบรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและการแจ้งเตือนผ่านไลน์

System for reporting problems using customer programs and notifications via LINE

สมนึก สินธุปวน[1]*, ชวัลวิทย์ เลาหาง[1], ณัฐชยา เตปันวงศ์[1], พุฒิพงศ์ พนมไพร[1],
ก่องกาญจน์ ดุลยไชย[1] และอรรถวิท ชังคมานนท์[1]
[1]สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้

Somnuek Sinthupuan[1]*, Chawanwit Laohang[1], Natchaya Tepanwong[1],
Puttipong Panomprai[1], Kongkarn Dullayachai[1] and Attawit Changkamanon[1]
[1]Computer Science Department, Faculty of Science, Maejo University
*ผู้นิพนธ์ประสานงาน: สมนึก สินธุปวน อีเมล: somnuk@mju.ac.th

Abstract:

Objectives: 1) Develop a system to receive reports of problems using customer programs and notify system administrators and employees via LINE, and 2) test system performance using Cypress automated testing frameworks. The system consists of five main parts: 1) constant data management for the system, 2) receiving problem notifications, 3) managing problems, 4) staff management and 5) project management. Automated system testing tests login and all main parts. Research results: Automated system testing was divided into five parts. It was found that the system was fully functional in all functions on the 5th time within the period specified according to the plan of 6 months, which was 40 percent faster than manual testing. The results of the research with the two companies found that a total of 120 problems were reported through the system, consisting of 60 problems from Company A and 60 problems from Company B. Problems that were resolved: Company A accounted for 70.0 percent, and Company B accounted for 70.0 percent. 84.0 percent.

Keyword: *Notification; Problem report; Line; Manual-Test; Automate-test*

บทคัดย่อ:

วัตถุประสงค์เพื่อ 1) พัฒนาระบบรับรายงานปัญหาการใช้งานโปรแกรมของลูกค้า และการแจ้งเตือนผู้ดูแลระบบและพนักงานผ่านไลน์ และ 2) เพื่อทดสอบประสิทธิภาพของระบบโดยใช้วิธีการทดสอบอัตโนมัติด้วยไซเพรสเฟรมเวิร์ค ซึ่งระบบประกอบด้วย 5 ส่วนประกอบหลัก ได้แก่ 1) การจัดการข้อมูลค่าคงที่สำหรับระบบ 2) การรับแจ้งปัญหา 3) การจัดการแก้ปัญหา 4) การจัดการพนักงาน และ 5) การจัดการโครงการ สำหรับการทดสอบระบบแบบอัตโนมัติ ได้ทำการทดสอบการเข้าสู่ระบบและส่วนประกอบหลักของระบบทั้งหมด โดยการวิจัยในการทดสอบระบบแบบอัตโนมัติได้แบ่งการทดสอบออกเป็น 5 ครั้ง ผลการวิจัยพบว่า ระบบทำงานได้ครบถ้วนทุกฟังก์ชันในครั้งที่ 5 ภายในระยะเวลาที่กำหนดตามแผน 6 เดือน ซึ่งเร็วกว่าการทดสอบด้วยมือร้อยละ 40 ผลการวิจัยกับบริษัททั้งสองบริษัทพบว่า ได้รับแจ้งรายงานปัญหาผ่านระบบจำนวนทั้งหมด 120 ปัญหา ประกอบด้วย บริษัททเอ จำนวน 60 ปัญหา และบริษัทบี จำนวน 60 ปัญหา ปัญหาที่ถูกแก้ไข บริษัทเอ คิดเป็นร้อยละ 70.0 และบริษัทบี คิดเป็นร้อยละ 84.0

คำสำคัญ: *การแจ้งเตือน; การรายงานปัญหา; ไลน์; การทดสอบระบบด้วยมือ; การทดสอบอัตโนมัติ*

1. บทนำ

บริษัทคราฟติ้งแล็บ จำกัด ดำเนินธุรกิจเกี่ยวกับการให้คำปรึกษาด้านการตลาดและการพัฒนาซอฟต์แวร์ตามความต้องการของลูกค้า ทางบริษัทได้พัฒนาระบบให้กับลูกค้าหลายโครงการและการทำบริการหลังการขาย รวมถึงรับการแจ้งปัญหาของลูกค้า ซึ่งลูกค้าส่วนใหญ่จะส่งปัญหาผ่าน Line แอปพลิเคชันทั่วไป หรือผ่านทางระบบ Email ของบริษัท โดยปัญหาที่เกิดขึ้นจากการแจ้งปัญหาผ่านซอฟต์แวร์ทั้ง 2 แบบคือ ข้อมูลการแจ้งปัญหาของลูกค้าเมื่อมีจำนวนมาก มีความยุ่งยากในการค้นหา การติดตาม การวิเคราะห์และการแก้ไขปัญหาสำหรับลูกค้า นอกจากนั้นการตอบกลับผ่านทางช่องทางแชทของ Line แอปพลิเคชันทั่วไป ไม่มีความต่อเนื่องกับปัญหาและไม่มีรายละเอียดที่เพียงพอสำหรับลูกค้า ในการอธิบายการแก้ปัญหาหรือข้อสงสัยทำให้แก้ปัญหาได้ไม่ถึงร้อยละ 50 ดังนั้น งานวิจัยจึงมีวัตถุประสงค์ เพื่อพัฒนาระบบรับรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและการแจ้งเตือนผู้ดูแลระบบและพนักงานผ่านไลน์ โดยระบบจะทำการรับรายงานปัญหาจากการใช้

งานของลูกค้า เก็บรวบรวมข้อมูลรายงานปัญหาเพื่อนำไปสู่การแก้ไขปัญหาสำหรับลูกค้า ซึ่งเป็นการเพิ่มประสิทธิภาพการให้บริการหลังการขาย และเพื่อลดระยะเวลาการพัฒนาระบบในส่วนการทดสอบระบบให้มีประสิทธิภาพเพิ่มขึ้นจากการทดสอบระบบแบบมือซึ่งบริษัทใช้เวลารวมการทดสอบประมาณ จำนวน 10 เดือน โดยใช้เวลาการพัฒนาระบบ จำนวน 6 เดือน ทดสอบและแก้ไขข้อผิดพลาดของโปรแกรมประมาณ จำนวน 4 เดือน งานวิจัยจึงมีวัตถุประสงค์ เพื่อทดสอบระบบแบบอัตโนมัติโดยใช้โปรแกรม Cypress ทำให้ผลการทดสอบระบบมีความรวดเร็วและมีความแม่นยำมากยิ่งขึ้น โดยใช้เวลา 6 เดือน โดยใช้เวลาการพัฒนาระบบ จำนวน 4 เดือน และทดสอบระบบพร้อมทั้งแก้ไข จำนวน 2 เดือน

2. วัตถุประสงค์

1) เพื่อพัฒนาระบบรับรายงานปัญหาการใช้งานโปรแกรมของลูกค้า และแจ้งเตือนผู้ดูแลระบบและพนักงานผ่านไลน์

2) เพื่อทดสอบประสิทธิภาพของระบบโดยใช้วิธีการทดสอบอัตโนมัติ

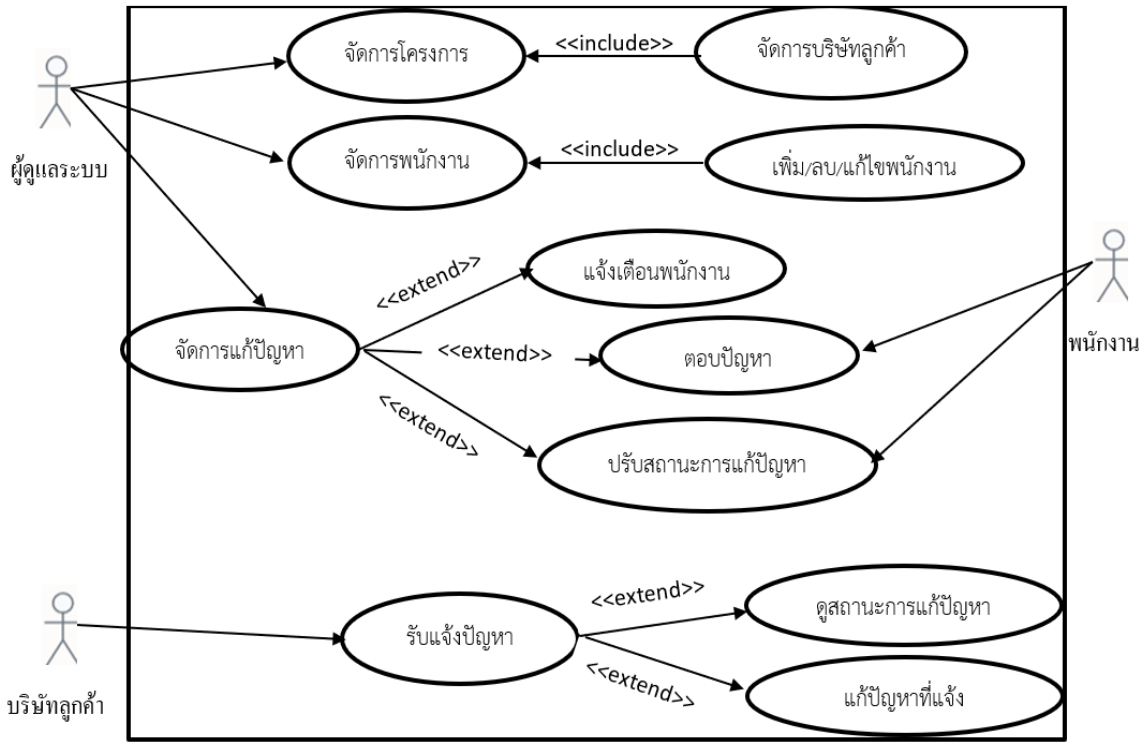
3. งานวิจัยที่เกี่ยวข้อง

ถนอม กองใจ และ อริษา ทาทอง (2565) ได้ทำการพัฒนาระบบแจ้งเตือนกิจกรรมและนัดหมายแบบอัตโนมัติผ่านแอปพลิเคชันไลน์ เพื่อส่งการแจ้งเตือนอัตโนมัติเมื่อมีกิจกรรมหรือการนัดหมายในแต่ละวัน ไปยังผู้ใช้งานผ่านแอปพลิเคชันไลน์บนโทรศัพท์มือถือ ทำให้ผู้ใช้ทราบล่วงหน้าและข้อมูลกิจกรรมจะถูกบันทึกไว้ ทำให้สามารถใช้ข้อมูลประกอบการวางแผนการจัดกิจกรรมล่วงหน้าและสรุปกิจกรรมในแต่ปีได้ ผลการวิจัยใช้การวัดผลเป็นความพึงพอใจจากผู้ใช้งานระบบประกอบด้วย ด้านความครบถ้วนของข้อมูลที่ส่งแจ้งเตือน ระดับมาก ด้านความสะดวกในการใช้งานระดับมากที่สุด ด้านความถูกต้องของข้อมูลในระดับ ระดับมากที่สุด ความพึงพอใจโดยรวม ระดับมากที่สุด และความพึงพอใจเฉลี่ยรวม ระดับมากที่สุด สาวิตรี วงษ์นุ่น (2563) ได้นำเสนอระบบการเข้าถึงข้อมูลในสถานการณ์ COVID -19 การทำงานของระบบประกอบด้วย 3 ส่วน ได้แก่ การรับค่าข้อมูล การประมวลผลเพื่อการส่งค่าข้อมูล และการแสดงผลสารสนเทศ รูปแบบการสื่อสารและแจ้งเตือนสารสนเทศใช้ Line Notify API เพื่อแก้ปัญหาการเข้าถึงข้อมูลในสถานการณ์ COVID -19 ผลการวิจัย

ใช้การวัดผลเป็นความพึงพอใจจากผู้ใช้งานระบบประกอบด้วย ด้านประสิทธิภาพของระบบ คะแนนเฉลี่ย 4.41 ด้านการออกแบบ คะแนนเฉลี่ย 4.63 และด้านกระบวนการ/ขั้นตอนการใช้งานระบบ คะแนนเฉลี่ย 4.50 อิมรอน แวมง อิสมาแอล พิทักษ์ลาวัลย์ วันอัฟฟาน มิมอ และพรรณี แพงทิพย์ (2563) ได้พัฒนาเว็บแอปพลิเคชันในการให้บริการถ่ายสำเนาเอกสารทางวิชาการ และแจ้งเตือนผ่านแอปพลิเคชันไลน์ ซึ่งการทำงานของระบบแบ่งออกเป็น 5 กลุ่มตามผู้ใช้ ได้แก่ ผู้ดูแลระบบ คณาจารย์ เจ้าหน้าที่ ฝ่ายวิชาการ ผู้อำนวยการสำนักงานคณบดีและเจ้าหน้าที่ถ่ายสำเนาเอกสาร ผลการวิจัยพบว่า ระบบสามารถทำงานได้หลายอุปกรณ์ ผู้ใช้งานสามารถขอใช้บริการถ่าย สำเนาได้ทุกที่ ทุกเวลา และสามารถติดตามเอกสารผ่านระบบได้ตลอดเวลา ผลการประเมินด้านการพัฒนาระบบ คะแนนเฉลี่ย 4.37 รัชพล แซ่เล้า เจนภพ ดำเนินผล ธนกฤษ จันทร์แสง ณสิทธิ์ เหล่าเส้น และพิทา จารุพูนผล (2564) ได้พัฒนาระบบการแจ้งเตือนการสูบบุหรี่ในพื้นที่ห้ามสูบในสถานศึกษาตามนโยบายของรัฐบาล ระบบประกอบด้วย เซ็นเซอร์ MQ2 Smoke Gas และกล้องวงจรปิด (IP Camera) เพื่อใช้สำหรับการตรวจจับการสูบบุหรี่ และส่งข้อความการแจ้งเตือนการสูบบุหรี่ รูปภาพและชื่อสถานที่ ผ่าน LINE Notify ผลการวิจัยพบว่า ระบบตรวจพบควันบุหรี่ 3 ระยะ ประกอบด้วย 20, 50 และ 100 เซนติเมตร ระบบสามารถตรวจพบได้เพียงระยะ คือ 20 และ 50 เซนติเมตรแต่ไม่สามารถตรวจพบได้ในระยะ 100 เซนติเมตร กฤษณะ พิริยะกิจไพบูลย์ (2547) ได้นำเสนอวิธีการสร้างข้อมูลสำหรับการทดสอบแบบอัตโนมัติ เพื่อช่วยผู้ทดสอบในการสร้างข้อมูลการทดสอบได้ง่ายขึ้น โดยเครื่องมือจะสุ่มการสร้างข้อมูลทดสอบตามเงื่อนไข ซึ่งจากการทดสอบการใช้งานเครื่องมือกับระบบงานจริง พบว่าเครื่องมือสามารถสร้างข้อมูลทดสอบได้ตรงกับคุณสมบัติของข้อมูลที่มีอยู่จริง พนารัตน์ เชี่ยววิทย์ (2545) ได้ทำการพัฒนาเครื่องมือการทดสอบซอฟต์แวร์ที่ทำการพัฒนาเชิงวัตถุโดยใช้ภาษาจาวา พบว่าภาษาจาวามีข้อจำกัดในการวิเคราะห์กรณีทดสอบ ดังนั้นเครื่องมือการทดสอบซอฟต์แวร์นี้ ควรมีการสร้างชุดข้อมูล สำหรับทดสอบโปรแกรมและมีการนำเข้าไปทดสอบโดยอัตโนมัติเพื่อให้เครื่องมือนี้มีความสมบูรณ์มากยิ่งขึ้น สันต์ทศน์ สุริยันต์ (2543) ได้นำการพัฒนาเครื่องมือสำหรับการวางแผนการทดสอบซอฟต์แวร์ 3 ระดับ ได้แก่ การทดสอบระดับหน่วย การทดสอบการรวม และการทดสอบระบบ เครื่องมือดังกล่าวสามารถเก็บรวบรวมข้อมูลที่เกิดจากการทดสอบโดยอิงกับมาตรฐานของเอกสารที่ใช้ในการทดสอบ และได้รับการทดสอบโดยใช้ข้อมูลจริงจากบริษัทพัฒนาซอฟต์แวร์ ธนพล สิชฌนุกฤษฎ์ (2544) ได้นำเสนอการพัฒนาระบบจัดการกรณีทดสอบซอฟต์แวร์ เพื่อนำมาช่วยผู้ทดสอบในการสร้างกรณีทดสอบ และเก็บข้อมูลกรณีทดสอบ ทำให้การ

ทดสอบทำได้สะดวกยิ่งขึ้น และเป็นการลดภาระของผู้ทดสอบอีกด้วย โดยการเก็บข้อมูลกรณีทดสอบและผลการทดสอบภายในระบบ Chai Phonbopit (2565) ได้นำเสนอระบบสมัครสมาชิกโดยใช้ภาษาและเฟรมเวิร์คเป็น React.js โดยมีการใช้งานร่วมกับ React Router ซึ่งมีการจัดการเกี่ยวกับความปลอดภัยของระบบผ่าน Firebase Authentication เพื่อทำหน้าที่ให้บริการในส่วนของการทำการ Routing ส่วน Prudchaya Palee (2562) ได้นำเสนอการสร้าง RESTful API โดยใช้บริการ Firebase เพื่อใช้สำหรับการเก็บข้อมูลผ่าน Cloud Storage โดยใช้ภาษา JavaScript ในการเขียนโปรแกรมร่วมกับเฟรมเวิร์คอื่นๆ ซึ่งสามารถทำงานบน NodeJS Chai Phonbopit (2563) ได้นำเสนอการกระจายซอฟต์แวร์บนเว็ปเซิร์ฟเวอร์ (Deployment) โดยผ่าน Firebase Hosting โดยพิจารณาจำนวนครั้งของการดาวน์โหลดซอฟต์แวร์และทำการจัดเก็บเป็นประวัติ (History) ของซอฟต์แวร์ดังกล่าว ทำให้การดาวน์โหลดมีหลายเวอร์ชันและผู้ใช้งานสามารถกลับไปใช้เวอร์ชั่นที่เก่ากว่าได้ (Rollback version) โดยงานวิจัยที่กล่าวมาข้างต้น มีความสอดคล้องกับวัตถุประสงค์ของงานวิจัยระบบรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและการแจ้งเตือนผ่านไลน์ โดยภาพรวมของระบบสามารถ แสดงได้ดังภาพที่ 1

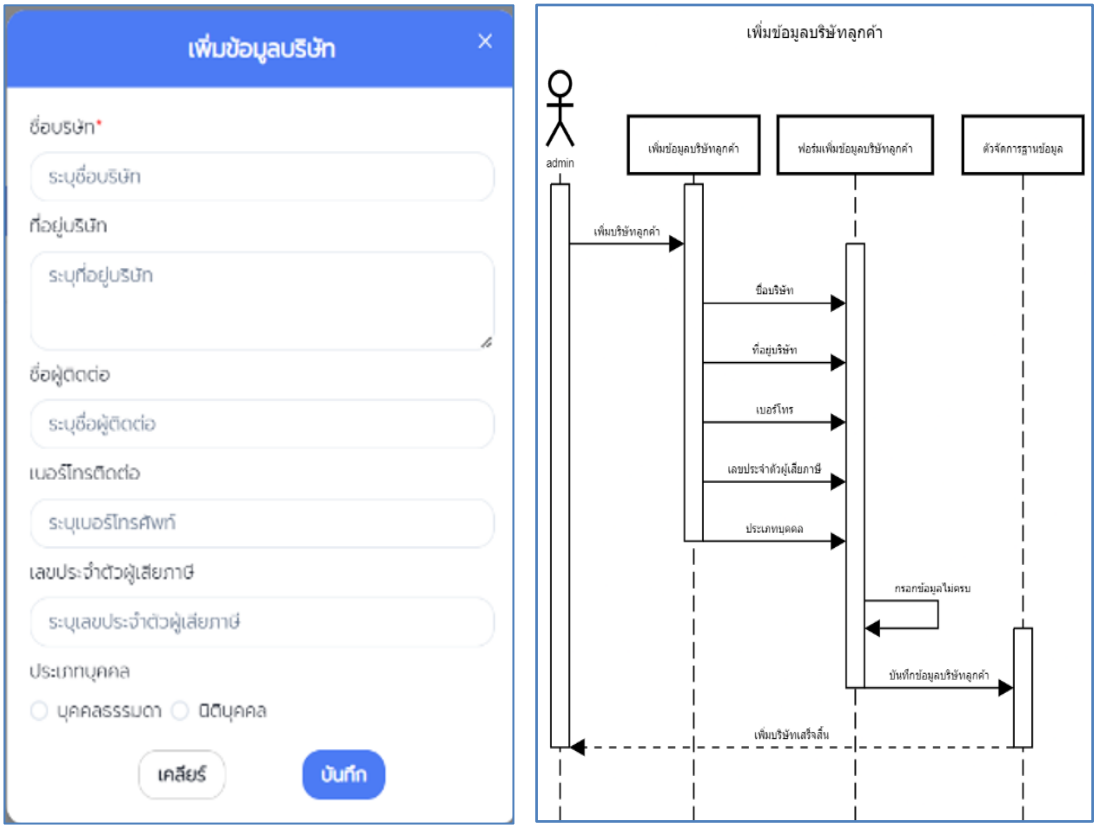
4. การพัฒนารายงานปัญหาการใช้งานโปรแกรมสำหรับลูกค้าโดยการแจ้งเตือนผ่านไลน์



ภาพที่ 1 ยูสเคสไดอะแกรมของ บริษัทลูกค้า ผู้ดูแลระบบ และพนักงาน

1) การเพิ่มข้อมูลค่าคงที่สำหรับระบบ

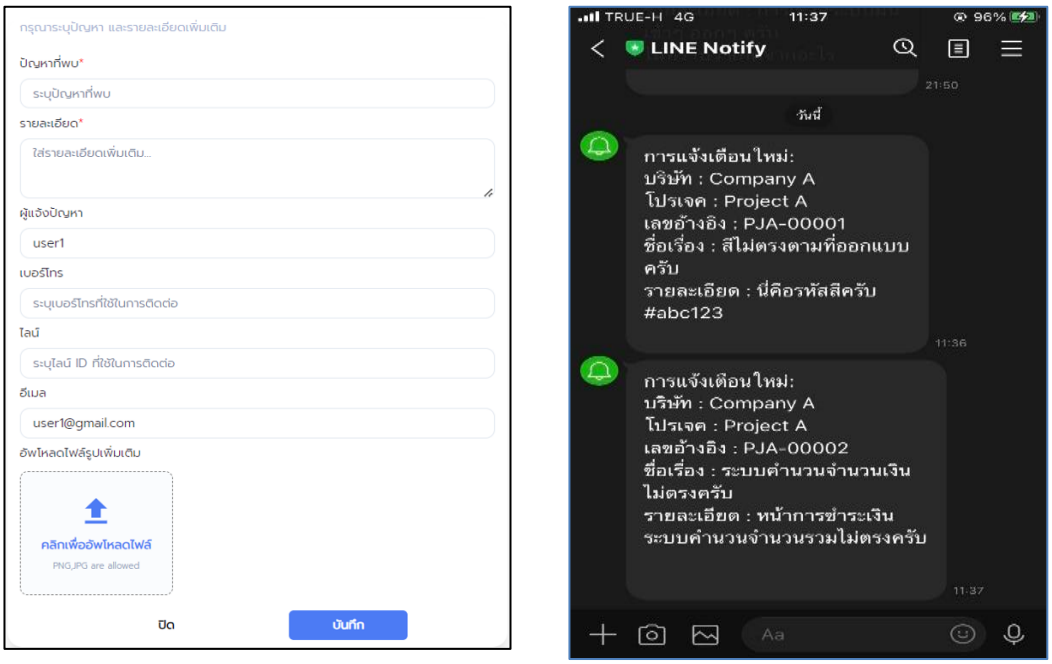
การเพิ่มข้อมูลค่าคงที่สำหรับระบบ สามารถดำเนินการเพิ่มข้อมูลบริษัทและข้อมูลโครงการ การเพิ่มข้อมูลบริษัทเข้าสู่ระบบ โดยผู้ดูแลระบบจะทำหน้าที่เพิ่มข้อมูลบริษัทก่อน และทำการเพิ่มข้อมูลโครงการ การเพิ่มข้อมูลของบริษัท สามารถทำได้โดย ผู้ดูแลระบบเข้าสู่หน้าฟอร์มเพิ่มข้อมูลบริษัทแล้วกรอกข้อมูลของบริษัทลูกค้า ซึ่งขั้นตอนการเพิ่มโครงการมีขั้นตอนการดำเนินงานเช่นเดียวกับการเพิ่มข้อมูลของบริษัท โดยบริษัทจะสามารถแจ้งปัญหาเข้ามาได้ หลังจากผู้ดูแลระบบทำการเพิ่มข้อมูลบริษัทและโครงการสำเร็จแล้วเท่านั้น การเพิ่มข้อมูลบริษัทแสดงได้ดังภาพที่ 2



ภาพที่ 2 ขั้นตอนการเพิ่มข้อมูลบริษัท

2) การรับแจ้งปัญหา

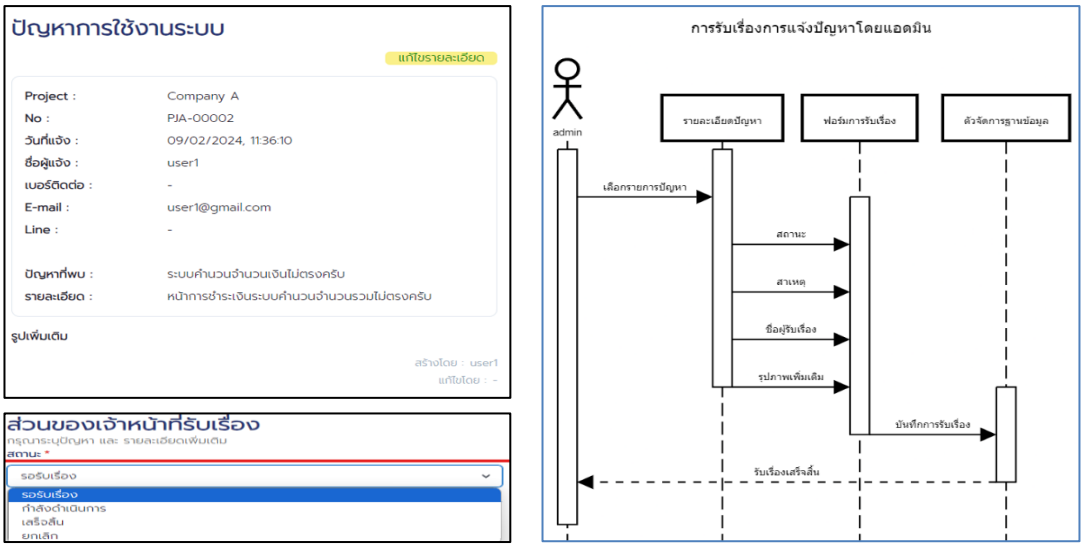
การรับแจ้งปัญหา บริษัทลูกค้าสามารถแจ้งปัญหาผ่านหน้าโครงการเว็บไซต์ของลูกค้า การแจ้งปัญหาการใช้งาน บริษัทจะกรอกปัญหาและรายละเอียดของปัญหาที่เกิดขึ้น เมื่อทำการบันทึกปัญหา ระบบจะมีการแจ้งเตือนผ่านไลน์ที่ได้รับการพัฒนาขึ้นไปยังผู้ดูแลระบบ โดยปัญหาที่บริษัทได้ทำการแจ้งปัญหาสามารถกลับมาแก้ไขในภายหลังได้แสดงได้ดังภาพที่ 3



ภาพที่ 3 แสดงการแจ้งปัญหา

3) การจัดการแก้ปัญหา

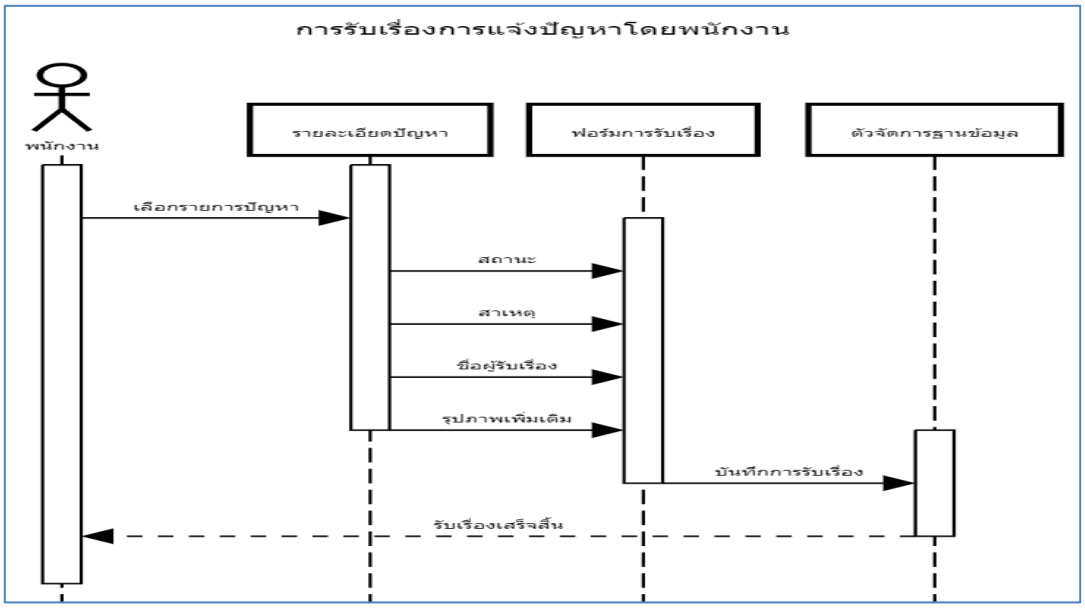
การจัดการแก้ปัญหา ผู้ดูแลระบบจะทำการเลือกบริษัทและโครงการที่มีการแจ้งปัญหาผ่านไลน์ ทำการตอบปัญหาหรือเสนอแนะแนวทางโดยละเอียดเกี่ยวกับวิธีการแก้ปัญหาด้วยตนเอง และเปลี่ยนสถานะปัญหา ซึ่งประกอบด้วย รอรับเรื่อง กำลังดำเนินการ เสร็จสิ้น และยกเลิก ในกรณีที่ไม่สามารถดำเนินการด้วยตนเองจะแจ้งเตือนไปยังพนักงานที่เกี่ยวข้องเพื่อดำเนินการต่อไป แสดงได้ดังภาพที่ 4



ภาพที่ 4 การตอบและการเปลี่ยนสถานะของปัญหาของผู้ดูแลระบบ

4) การเพิ่มพนักงานแก้ปัญหา

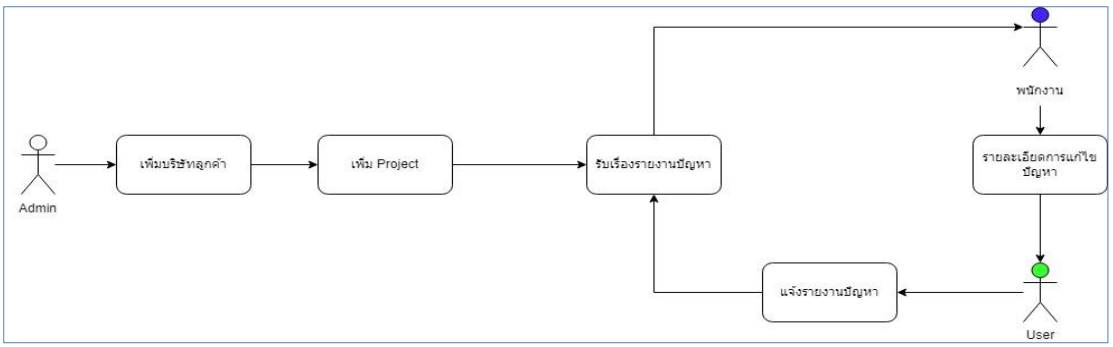
การเพิ่มพนักงานแก้ปัญหา ในกรณีที่ผู้ดูแลระบบเห็นว่ามีพนักงานที่เกี่ยวข้องกับปัญหา ผู้ดูแลระบบจะสามารถเพิ่มพนักงาน โดยการกรอกข้อมูลพนักงาน ซึ่งพนักงานที่ได้รับการเพิ่มข้อมูลสามารถทำขั้นตอนการตอบปัญหาและปรับสถานะการแก้ปัญหาได้แสดงได้ดังภาพที่ 5



ภาพที่ 5 การตอบและการเปลี่ยนสถานะของปัญหาของพนักงานที่เกี่ยวข้อง

5. การทดสอบระบบแบบอัตโนมัติ

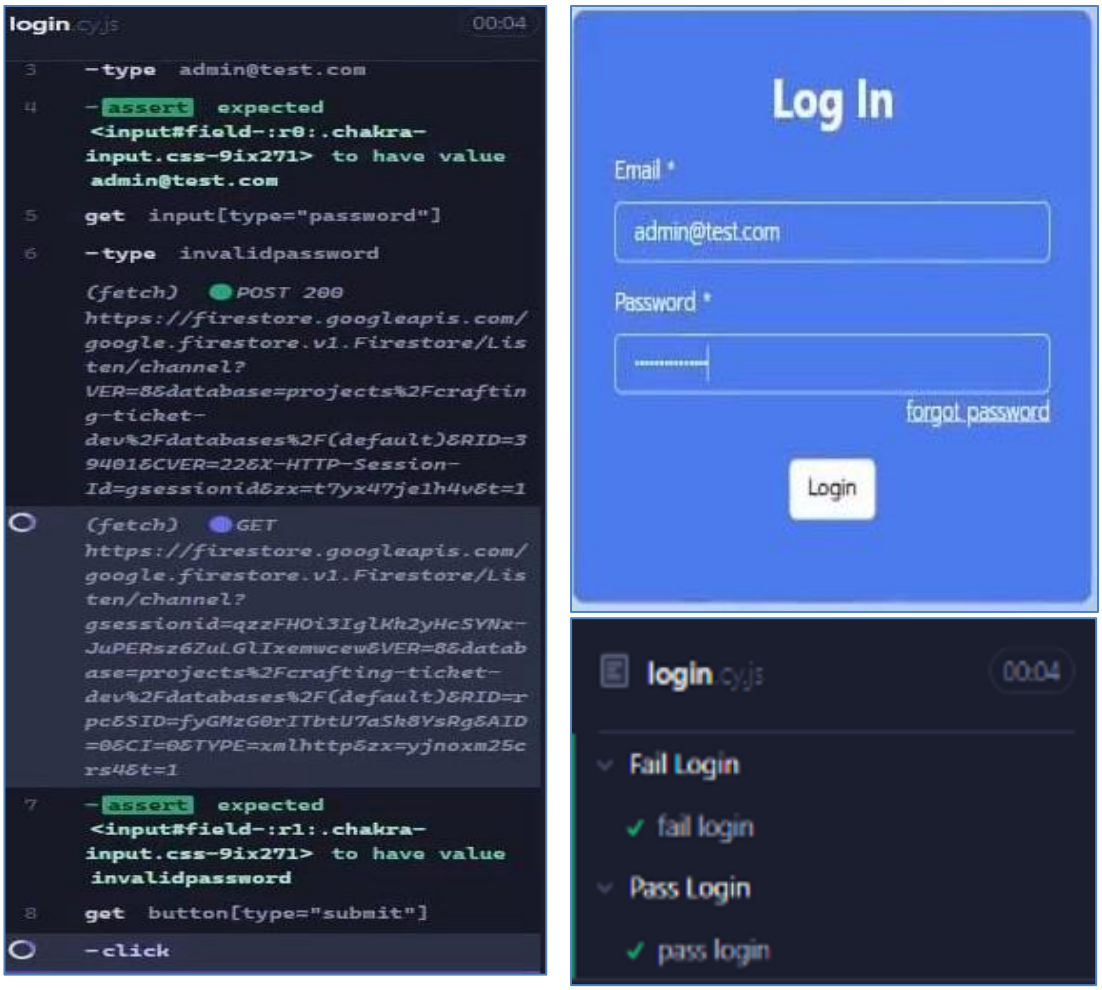
ในขั้นตอนการทดสอบระบบแบบอัตโนมัติได้มีการออกแบบ Flow การทำงานของระบบ มีขั้นตอนการทำงานของระบบ โดยเริ่มจาก Admin เพิ่มบริษัทลูกค้าและเพิ่ม Project ของบริษัทลูกค้า จากนั้นเมื่อลูกค้าพบปัญหาจากการใช้งานจะแจ้งปัญหาเข้ามาในระบบ Admin จะเข้ามารับเรื่องรายงานปัญหาและตอบปัญหาของลูกค้า หากเป็นปัญหาที่ Admin ไม่สามารถแก้ไขได้ จะส่งต่อไปยังพนักงานที่ดูแลในส่วนที่เกี่ยวข้องและทำการแก้ไข เมื่อพนักงานแก้ไขเสร็จสิ้น พนักงานจะเข้ามาเปลี่ยนสถานะรายงานปัญหาและแจ้งสาเหตุและรายละเอียดการแก้ไขให้กับลูกค้าแสดงได้ดังภาพที่ 6



ภาพที่ 6 Flow การทำงานของระบบ

1) การทดสอบการเข้าสู่ระบบ

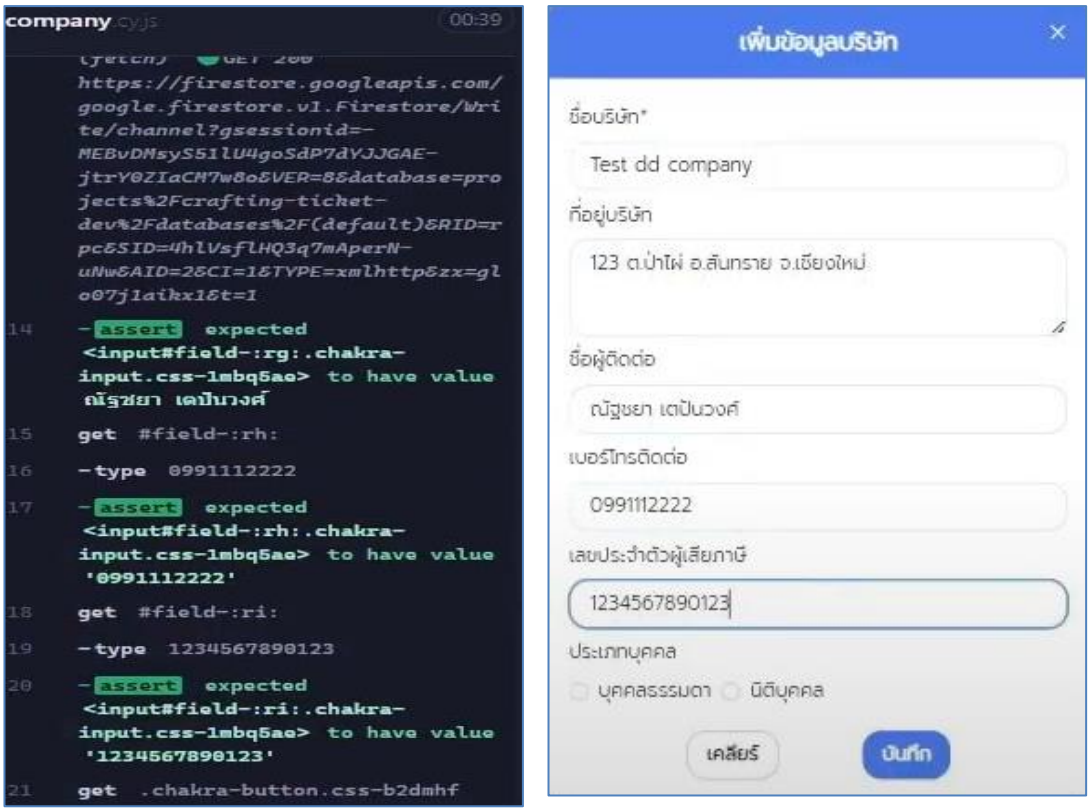
การเข้าสู่ระบบ ในการทดสอบการเข้าสู่ระบบเริ่มต้นจากเข้าสู่หน้าเข้าสู่ระบบ จากนั้นกรอก Email และ Password มีการตรวจสอบการกรอกข้อมูลว่าถูกกรอกลงในช่องหรือไม่ จากนั้นกดปุ่ม Login ถ้า Email หรือ Password ถูกต้องจะตรวจสอบว่ามีข้อความ Success Message แสดงขึ้นมา และ ถ้า Email หรือ Password ไม่ถูกต้องจะตรวจสอบว่ามีข้อความ Error Message แสดงขึ้นมา แสดงได้ดังภาพที่ 7



ภาพที่ 7 การทดสอบการกรอก Email และ Password

2) การทดสอบการเพิ่มค่าคงที่ (การเพิ่มข้อมูลบริษัท)

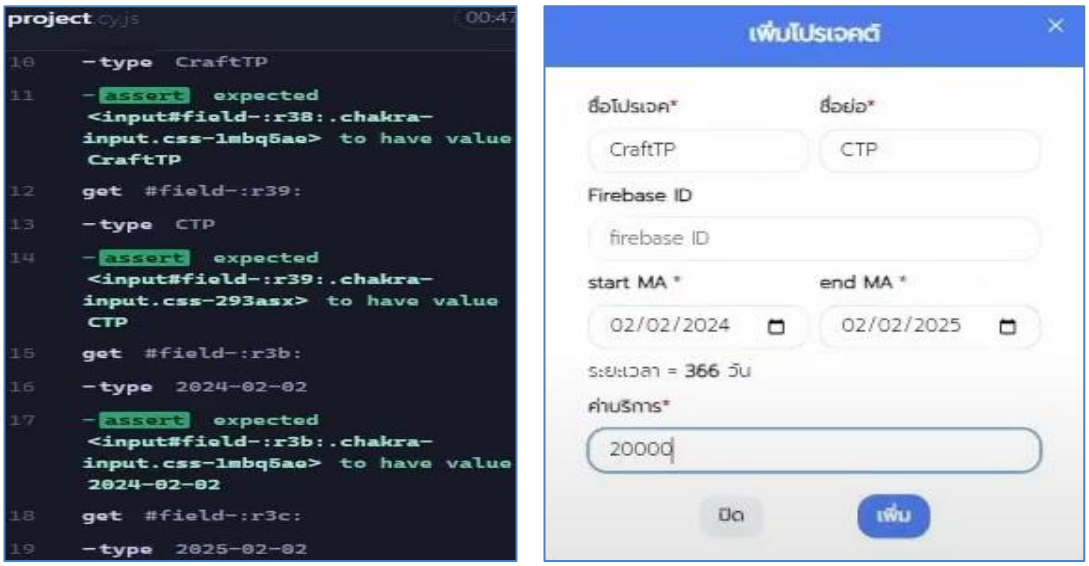
การเพิ่มข้อมูลบริษัท การเพิ่มข้อมูลบริษัทจะเริ่มจากเข้าสู่หน้าข้อมูลลูกค้า และกดปุ่มเพิ่มข้อมูลบริษัท จะมีฟอร์มเพิ่มข้อมูลบริษัทแสดงขึ้นมา จากนั้นกรอกข้อมูลบริษัทลูกค้าลงในฟอร์มเพิ่มข้อมูลบริษัท จะมีการตรวจสอบว่าข้อมูล ถูกกรอกลงในแต่ละช่องหรือไม่ และทำการกดบันทึก จะมีการตรวจสอบว่าข้อมูลที่บันทึกลงไปมีในตารางข้อมูลลูกค้า แสดงได้ดังภาพที่ 8



ภาพที่ 8 การทดสอบการเพิ่มข้อมูลบริษัท

3) การทดสอบการเพิ่มค่าคงที่ (การเพิ่มโครงการ)

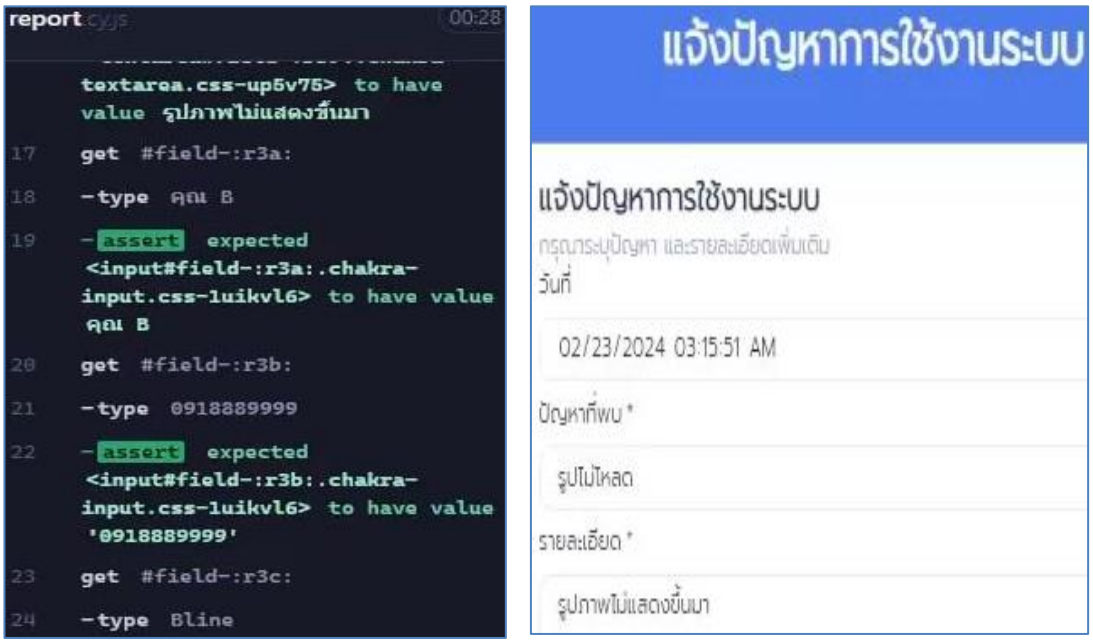
การเพิ่มโครงการ เริ่มจากเข้าสู่หน้าข้อมูล Projects และกดปุ่ม เพิ่ม project ระบบจะแสดงฟอร์ม เพิ่มโครงการขึ้นมา จากนั้นกรอกข้อมูลโปรเจกต์ลงในฟอร์มและตรวจสอบว่าข้อมูลสามารถกรอกลงในฟอร์มได้ หรือไม่ จากนั้นกดปุ่ม เพิ่ม และตรวจสอบว่าข้อมูลที่เพิ่ม มีในตารางข้องมูล Projects แสดงได้ดังภาพที่ 9



ภาพที่ 9 ทดสอบการเพิ่มโปรเจกต์

4) การทดสอบการแจ้งรายงานปัญหา

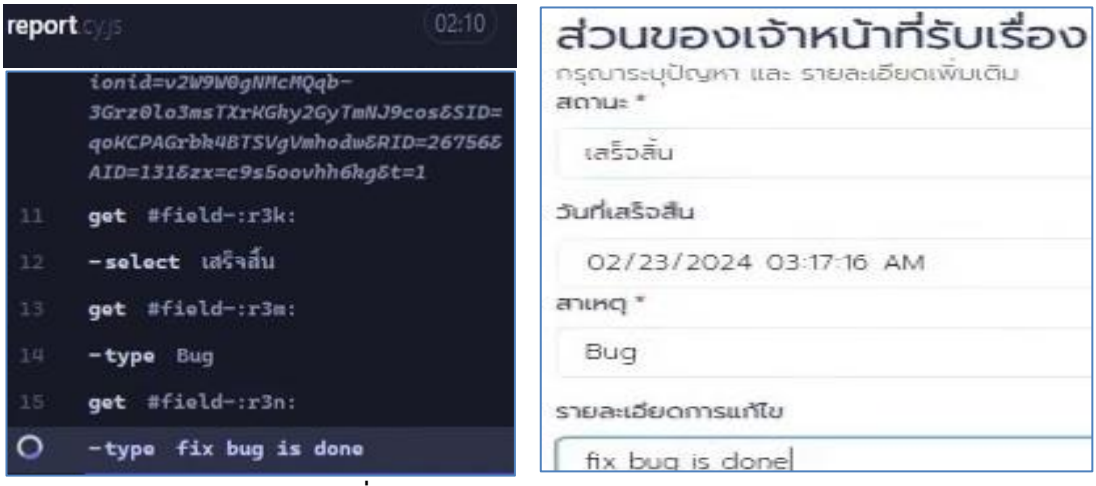
การแจ้งรายงานปัญหา เริ่มจากเข้าสู่หน้าแจ้งปัญหาการใช้งานระบบ และทำการกรอกข้อมูลรายงานปัญหา และตรวจสอบการกรอกข้อมูล จากนั้นกดบันทึกและตรวจสอบว่าข้อมูลรายงานปัญหาถูกบันทึกลงในตารางรายงานปัญหา แสดงได้ดังภาพที่ 10



ภาพที่ 10 ทดสอบการแจ้งรายงานปัญหา

5) การทดสอบการตอบรายงานปัญหา

ตอบรายงานปัญหา เริ่มจากเข้าสู่หน้าแจ้งปัญหาการใช้งานระบบ และเลือกรายงานปัญหาที่ต้องการตอบ จากนั้นทำการแก้ไขสถานะและกรอกข้อมูลสาเหตุ รายละเอียดปัญหาและตรวจสอบการกรอกข้อมูล จากนั้นกดบันทึก และตรวจสอบว่าสถานะรายงานปัญหาถูกแก้ไขในตารางรายงานปัญหา แสดงได้ดังภาพที่ 11



ภาพที่ 11 ตรวจสอบการตอบรายงานปัญหา

6) การทดสอบตั้งแต่เริ่มต้นจนจบการทำงาน

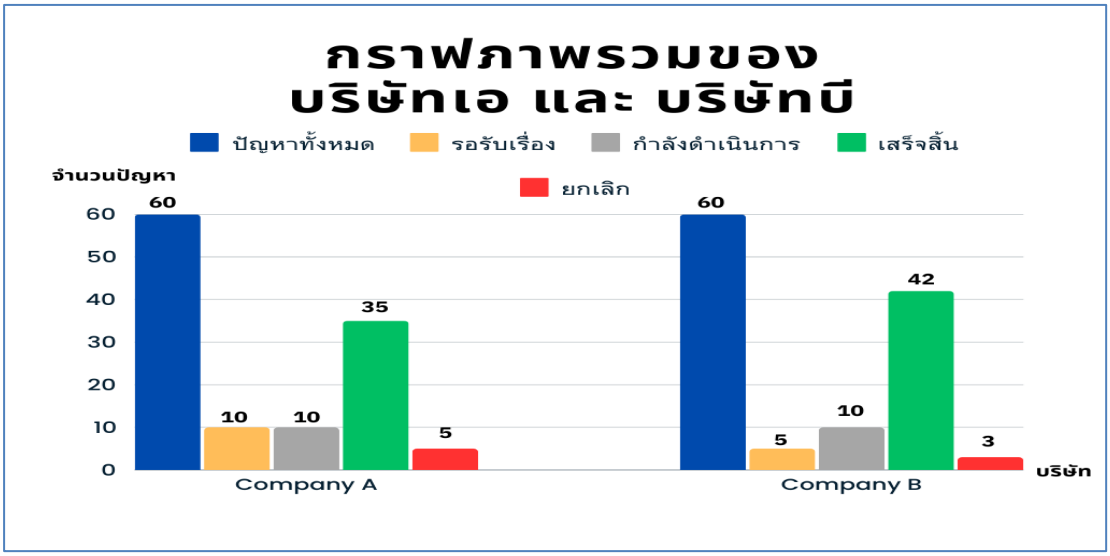
การทดสอบตั้งแต่เริ่มต้น Flow จนจบการทำงาน จะทดสอบตั้งแต่การเข้าสู่ระบบ Admin เพิ่มบริษัทลูกค้า และเพิ่ม Project ของบริษัทลูกค้า จากนั้นแจ้งปัญหาเข้ามาในระบบ และตอบรายงานปัญหาทั้ง 3 สถานะ แสดงได้ดังภาพที่ 12

ภาพที่ 12 ตรวจสอบการตั้งแต่เริ่มต้นจนจบการทำงาน

6. ผลการวิจัย

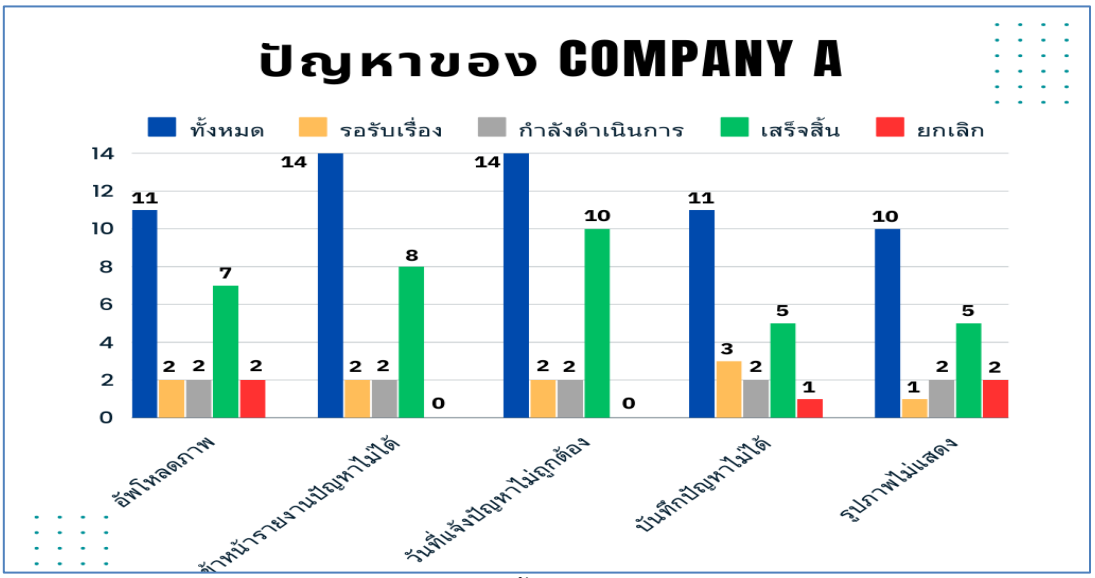
1) ผลการพัฒนาระบบรับรายงานปัญหาการใช้งานโปรแกรมของลูกค้า และการแจ้งเตือนผู้ดูแลระบบและพนักงานผ่านไลน์

การพัฒนาระบบได้ทำการทดสอบการทำงานของระบบในระยะเวลา 6 เดือน (การทดสอบระบบรอบสุดท้าย) โดยกำหนดให้มีผู้ดูแลระบบทำหน้าที่รับเรื่อง จำนวน 1 คน พนักงาน จำนวน 4 คน และบริษัทลูกค้า จำนวน 2 บริษัท ซึ่งได้รับซอฟต์แวร์ระบบเดียวกัน เพื่อทดสอบการรับเรื่องร้องเรียนเกี่ยวกับปัญหาของระบบและการแจ้งเตือนผ่านไลน์ บริษัทเอ และบริษัทบี เป็นบริษัทเข้าร่วมการทดสอบระบบกับบริษัทจริง แต่เนื่องจากทั้ง 2 บริษัท เป็นบริษัทลูกค้า ทางบริษัทจึงมีความจำเป็นต้องสงวนชื่อจริงของบริษัทไว้เป็นความลับของบริษัท เพื่อป้องกันความเสียหายที่อาจจะเกิดขึ้นกับบริษัทซึ่งมีสถานะเป็นลูกค้าของบริษัท ดังนั้นการรายงานผลของงานวิจัยจึงได้สมมุติชื่อขึ้น



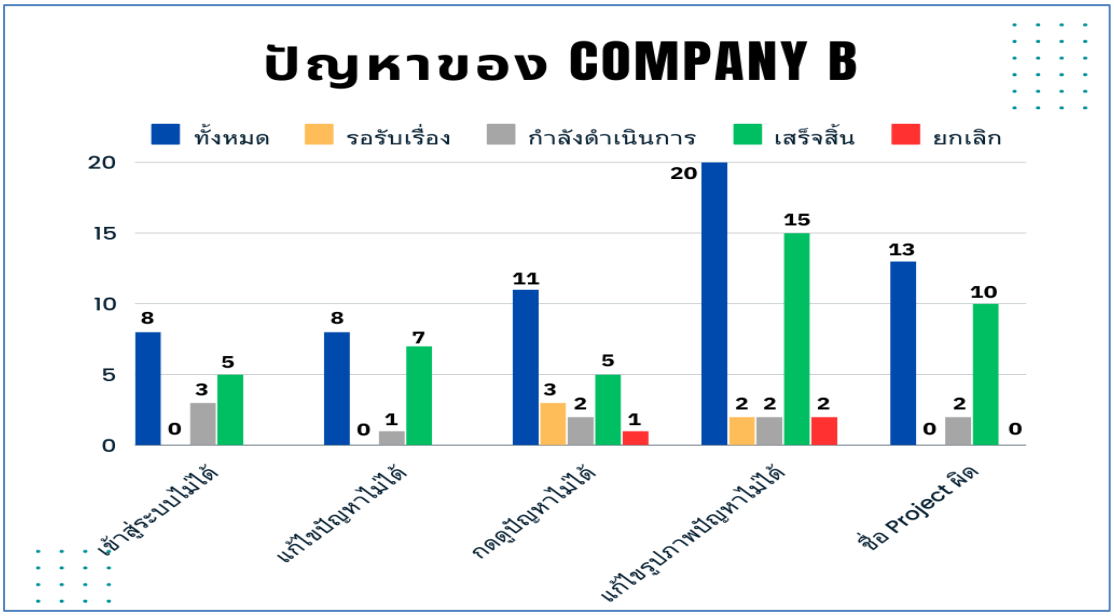
ภาพที่ 13 ภาพรวมการรายงานปัญหา

บริษัท A พบว่า ปัญหาที่อยู่ในสถานะเสร็จสิ้น จำนวน 35 ปัญหา (ร้อยละ 70) รอดำเนินการ จำนวน 10 ปัญหา และกำลังดำเนินการ จำนวน 10 ปัญหา สถานะปัญหาที่ถูกยกเลิกแล้ว จำนวน 5 ปัญหา **บริษัท B พบว่า** ปัญหาที่อยู่ในสถานะเสร็จสิ้น จำนวน 42 ปัญหา (ร้อยละ 84) สถานะรอดำเนินการ จำนวน 10 ปัญหา สถานะรอรับเรื่อง จำนวน 5 ปัญหา และ สถานะยกเลิก จำนวน 3 ปัญหา สรุปแล้วแต่ละบริษัทมีการแจ้งปัญหาทั้งหมด 60 ปัญหา จากการเปรียบเทียบของ 2 บริษัท พบว่า บริษัทบีมีการแก้ไขปัญหาเสร็จสิ้นมากกว่าบริษัทเอ รวมถึงปัญหาที่อยู่ในสถานะรอรับเรื่องและสถานะรอดำเนินการ บริษัทบีมีจำนวนปัญหาน้อยกว่าบริษัทเอ แสดงได้ดังภาพที่ 13



ภาพที่ 14 แสดงสถานะปัญหาทั้งหมดของแต่ละปัญหาของบริษัท A

บริษัท A ปัญหาเข้าหน้ารายงานไม่ได้และปัญหาวันที่แจ้งปัญหาไม่ถูกต้อง มีการแจ้งเข้าสู่ระบบมากที่สุด จำนวน 14 ปัญหา ปัญหารูปภาพไม่แสดง มีการแจ้งน้อยที่สุด จำนวน 10 ปัญหา ปัญหาที่มีการแก้ไขมากที่สุดคือ ปัญหาวันที่แจ้งปัญหาไม่ถูกต้อง จำนวน 10 ปัญหา และปัญหาบันทึกปัญหาไม่ได้ และปัญหารูปภาพไม่แสดง มีการแก้ไขน้อยที่สุด จำนวน 5 ปัญหา แสดงได้ดังภาพที่ 14



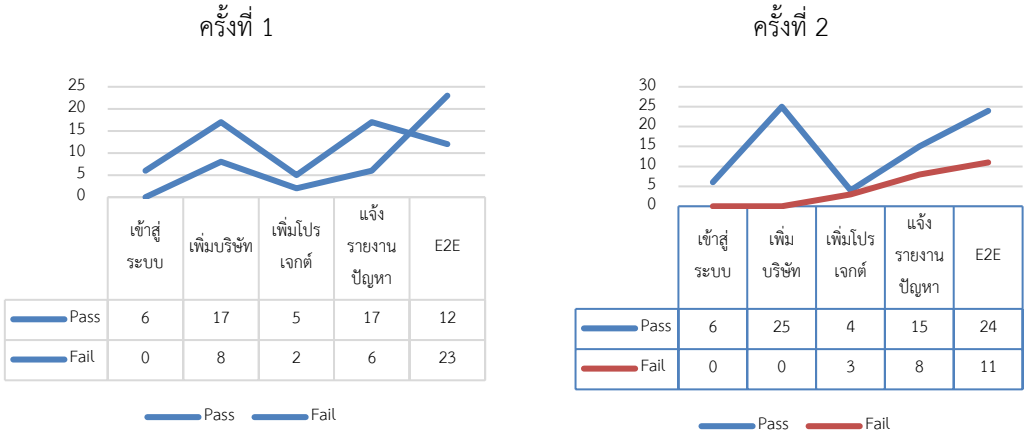
ภาพที่ 15 แสดงสถานะปัญหาทั้งหมดของแต่ละปัญหาของ Company B

บริษัท B พบว่า ปัญหาการแก้ไขรูปภาพไม่ได้ มีการแจ้งเข้าสู่ระบบมากที่สุด จำนวน 20 ปัญหา ปัญหาเข้าสู่ระบบไม่ได้ และการแก้ไขปัญหาไม่ได้ มีการแจ้งน้อยที่สุด จำนวน 8 ปัญหา ส่วนปัญหาที่มีการแก้ไขมากที่สุดคือ ปัญหาแก้ไขรูปภาพปัญหาไม่ได้ จำนวน 15 ปัญหา และปัญหาที่มีการแก้ไขน้อยที่สุดคือปัญหาเข้าสู่ระบบไม่ได้ จำนวน 5 ปัญหา แสดงได้ดังภาพที่ 15

2) ผลการทดสอบระบบแบบอัตโนมัติ

การทดสอบระบบแบบอัตโนมัติ จากผลการทดสอบระบบแบบอัตโนมัติ โดยกำหนดหัวข้อการทดสอบ 5 หัวข้อ ได้แก่ เข้าสู่ระบบ เพิ่มบริษัท เพิ่มโปรเจกต์ แจ้งรายงานปัญหา และการทดสอบการทำงานตั้งแต่เริ่มต้นจนจบการทำงาน ซึ่งกำหนดแผนการพัฒนาและแผนการทดสอบระบบให้เสร็จสิ้นภายในระยะเวลา 6 เดือน (ฝึกงาน จำนวน 2 เดือน และฝึกสหกิจศึกษา จำนวน 4 เดือน) โดยนักศึกษาจะต้องพัฒนาโปรแกรมให้เสร็จสิ้นภายใน 4 เดือน (เริ่มทดสอบระบบพร้อมปรับปรุงระบบจำนวน 1 ครั้ง) เดือนที่ 5 (ทดสอบกลางเดือน จำนวน 1 ครั้ง และปลายเดือน จำนวน 1 ครั้ง) เดือนที่ 6 ซึ่งเป็นเดือนสุดท้าย (ทดสอบกลางเดือน จำนวน 1 ครั้งและปลายเดือน จำนวน 1 ครั้ง) พบว่า โดย

การทดสอบทุกครั้งจะทดสอบทั้ง 5 หัว เหมือนกันทุกครั้ง เพื่อประเมินความพร้อมก่อนการนำโปรแกรมไปให้ลูกค้าใช้งานจริง



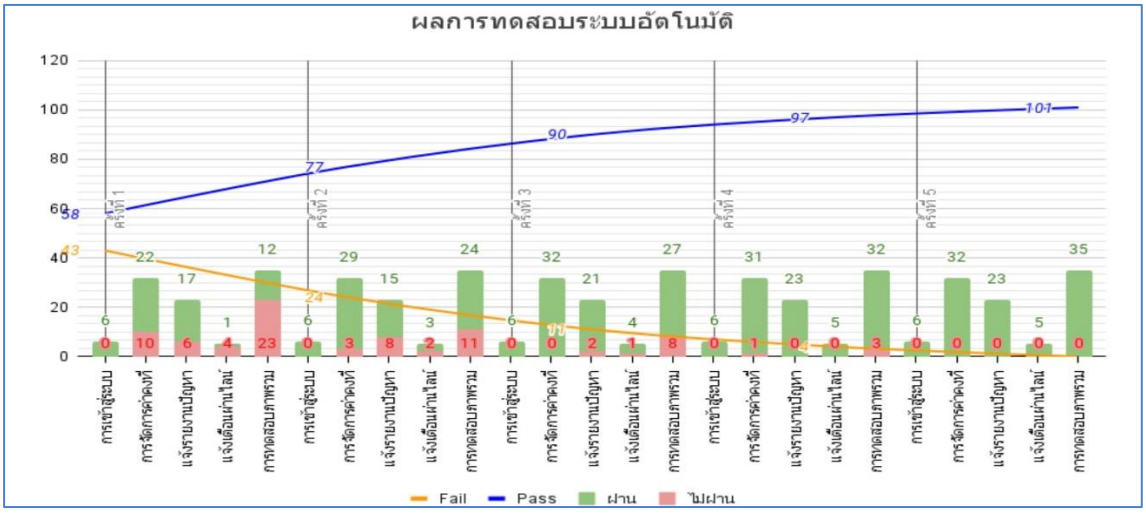
ภาพที่ 16 การทดสอบระบบอัตโนมัติ ครั้งที่ 1 และ 2

จากภาพที่ 16 เป็นการทดสอบอัตโนมัติ **ครั้งที่ 1)** ประกอบด้วยหัวข้อการทดสอบระบบได้แก่ 1) เข้าสู่ระบบ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 6 ครั้ง ผ่าน จำนวน 6 ครั้ง 2) เพิ่มบริษัท มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 25 ครั้ง ผ่าน จำนวน 17 ครั้ง ไม่ผ่าน จำนวน 8 ครั้ง 3) เพิ่มโปรเจกต์ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 7 ครั้ง ผ่าน จำนวน 5 ครั้ง ไม่ผ่าน จำนวน 2 ครั้ง 4) แจ้งรายงานปัญหา มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 23 ครั้ง ผ่าน จำนวน 17 ครั้ง ไม่ผ่าน จำนวน 6 ครั้ง และ 5) การทดสอบการทำงานตั้งแต่เริ่มต้น flow จนจบการทำงาน มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 35 ครั้ง ผ่าน จำนวน 12 ครั้ง ไม่ผ่าน จำนวน 23 ครั้ง **ครั้งที่ 2)** ประกอบด้วยหัวข้อการทดสอบระบบได้แก่ 1) เข้าสู่ระบบ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 6 ครั้ง ผ่าน จำนวน 6 ครั้ง 2) เพิ่มบริษัท มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 25 ครั้ง ผ่าน จำนวน 25 ครั้ง 3) เพิ่มโปรเจกต์ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 7 ครั้ง ผ่าน จำนวน 4 ครั้ง ไม่ผ่าน จำนวน 3 ครั้ง 4) แจ้งรายงานปัญหา มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 23 ครั้ง ผ่าน จำนวน 15 ครั้ง ไม่ผ่าน จำนวน 8 ครั้ง และ 5) การทดสอบการทำงานตั้งแต่เริ่มต้น flow จนจบการทำงาน มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 35 ครั้ง ผ่าน จำนวน 24 ครั้ง ไม่ผ่าน จำนวน 11 ครั้ง

การทดสอบอัตโนมัติ **ครั้งที่ 3)** ประกอบด้วยหัวข้อการทดสอบระบบได้แก่ 1) เข้าสู่ระบบ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 6 ครั้ง ผ่าน จำนวน 6 ครั้ง 2) เพิ่มบริษัท มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 25 ครั้ง ผ่าน จำนวน 25 ครั้ง 3) เพิ่มโปรเจกต์ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 7 ครั้ง ผ่าน จำนวน 7 ครั้ง 4) แจ้งรายงานปัญหา มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 23 ครั้ง ผ่าน จำนวน 21 ครั้ง ไม่ผ่าน จำนวน 2 ครั้ง และ 5)

การทดสอบการทำงานตั้งแต่เริ่มต้น flow จนจบการทำงาน มีการใช้ Should ในการทดสอบทั้งหมด 35 ครั้ง ผ่าน จำนวน 27 ครั้ง ไม่ผ่าน จำนวน 8 ครั้ง **ครั้งที่ 4)** ประกอบด้วยหัวข้อการทดสอบระบบ ได้แก่ 1) เข้าสู่ระบบ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 6 ครั้ง ผ่าน จำนวน 6 ครั้ง 2) เพิ่มบริษัท มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 25 ครั้ง ผ่าน จำนวน 25 ครั้ง 3) เพิ่มโปรเจกต์ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 7 ครั้ง ผ่าน จำนวน 6 ครั้ง ไม่ผ่าน จำนวน 1 ครั้ง 4) แจ้งรายงานปัญหา มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 23 ครั้ง ผ่าน จำนวน 23 ครั้ง และ 5) การทดสอบการทำงานตั้งแต่เริ่มต้น flow จนจบการทำงาน มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 35 ครั้ง ผ่าน จำนวน 32 ครั้ง ไม่ผ่าน จำนวน 3 ครั้ง **ครั้งที่ 5)** ประกอบด้วยหัวข้อการทดสอบระบบได้แก่ 1) เข้าสู่ระบบ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 6 ครั้ง ผ่าน จำนวน 6 ครั้ง 2) เพิ่มบริษัท มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 25 ครั้ง ผ่าน จำนวน 25 ครั้ง 3) เพิ่มโปรเจกต์ มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 7 ครั้ง ผ่าน จำนวน 7 ครั้ง 4) แจ้งรายงานปัญหา มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 23 ครั้ง ผ่าน จำนวน 23 ครั้ง และ 5) การทดสอบการทำงานตั้งแต่เริ่มต้น flow จนจบการทำงาน มีการใช้ Should ในการทดสอบทั้งหมด จำนวน 35 ครั้ง ผ่าน จำนวน 35 ครั้ง แสดงได้ดังภาพที่ 17

ภาพที่ 17 การทดสอบระบบอัตโนมัติ ครั้งที่ 3 - 5



ภาพที่ 18 ผลการทดสอบระบบอัตโนมัติ

7. บทสรุปและอภิปรายผลการวิจัย

จากกราฟการทดสอบระบบแบบอัตโนมัติ จะเห็นได้ว่าการทดสอบระบบแบบอัตโนมัติทั้งหมด 5 ครั้ง ในครั้งที่ 1 จนถึง ครั้งที่ 4 พบว่ามีการทดสอบแล้วพบปัญหาของระบบน้อยลงตามลำดับ และในครั้งที่ 5 สามารถทดสอบแล้วไม่พบปัญหาของระบบ จากภาพที่ 18 ผลการทดสอบระบบอัตโนมัติ ซึ่งตรงตามแผนที่บริษัทวางไว้โดยนักศึกษาที่ฝึกงานและฝึกสหกิจศึกษาจะต้องดำเนินการพัฒนาโปรแกรมและทดสอบโดยใช้ระบบอัตโนมัติให้แล้วเสร็จภายใน 6 เดือน ซึ่งในการพัฒนาโปรแกรมของบริษัทจะใช้เวลา 6 เดือนสำหรับการพัฒนาและทดสอบระบบด้วยมือใช้เวลา 2 เดือน และแก้ไขระบบอีก 2 เดือน รวมเป็น 10 เดือน ดังนั้นการทดสอบระบบแบบอัตโนมัติจึงเป็นวิธีที่ทำการให้พัฒนาโปรแกรมและการทดสอบโปรแกรมทำได้อย่างรวดเร็วขึ้น ซึ่งสอดคล้องกับงานวิจัยของ สันต์ทศน์ สุริยันต์ (2543) โดยงานวิจัยนี้ได้ทำการทดสอบกับข้อมูลจริงของบริษัทลูกค้า และแบ่งการทดสอบออกเป็น 2 ส่วน ประกอบด้วย การทดสอบระดับหน่วย และการทดสอบรวม โดยส่งผลทำให้ประสิทธิภาพการทดสอบระบบของบริษัทเพิ่มขึ้นเมื่อเปรียบเทียบกับการทดสอบระบบด้วยมือ

ผลการวิจัยกับบริษัททั้งสองบริษัทพบว่า ได้รับแจ้งรายงานปัญหาผ่านระบบจำนวนทั้งหมด 120 ปัญหา ประกอบด้วย บริษัททเอ จำนวน 60 ปัญหา และบริษัทบี จำนวน 60 ปัญหา ปัญหาที่ถูกแก้ไข บริษัทเอ คิดเป็นร้อยละ 70.0 และบริษัทบี คิดเป็นร้อยละ 84.0 ซึ่งมากกว่าระบบเดิมซึ่งแก้ปัญหาร้อยละ 50.0 ซึ่งสอดคล้องกับงานวิจัยของ ถนอม กองใจ และ อริษา ทาทอง (2565) ที่ได้ผลการวิจัยใช้การวัดผลเป็นความพึงพอใจจากผู้ใช้งานระบบ ประกอบด้วย ด้านความครบถ้วนของข้อมูลที่ส่งแจ้งเตือน ระดับมาก ด้านความสะดวกในการใช้งาน ระดับมากที่สุด ด้านความถูกต้องของข้อมูลในระดับ แต่งานวิจัยนี้ใช้ประเมินความสำเร็จของงานวิจัยจากการแก้ปัญหาได้จริงของบริษัททั้งสองบริษัทคิดเป็นร้อยละ 77 ซึ่งสูงกว่าระบบเดิมเนื่องจากความครบถ้วนของข้อมูลที่ส่งแจ้งเตือน

8. เอกสารอ้างอิง

กฤษณะ พิริยะกิจไพบูลย์. (2547). **การพัฒนาเครื่องมือสร้างข้อมูลทดสอบเพื่อสนับสนุนการทดสอบซอฟต์แวร์จากสคีมาของฐานข้อมูล**. (วิทยานิพนธ์ ปริญญามหาบัณฑิต). จุฬาลงกรณ์มหาวิทยาลัย.

ถนอม กองใจ และ อริษา ทาทอง. (2565). การพัฒนาระบบแจ้งเตือนกิจกรรมและการนัดหมายอัตโนมัติผ่านแอปพลิเคชันไลน์.**วารสาร Mahidol R2R e-Journal** 9(2): 32 - 45.

ธนพล สิชฌนุกฤษฎ์. (2544). **การพัฒนาระบบจัดการกรณีทดสอบซอฟต์แวร์**. (วิทยานิพนธ์ปริญญามหาบัณฑิต). จุฬาลงกรณ์มหาวิทยาลัย.

พนารัตน์ เชี่ยววิทย์. (2545). **การพัฒนาเครื่องมือการทดสอบซอฟต์แวร์ในระดับหน่วยสำหรับกระบวนการพัฒนาเชิงวัตถุ**. (วิทยานิพนธ์ ปริญญามหาบัณฑิต). มหาวิทยาลัยมหิดล.

รัชพล แซ่เล้า เจนภพ ดาเนินผล ธนกฤษ จันทร์แสง ณสิทธ์ เหล่าเส็น และพิทา จารุพูนผล. (2564). **ระบบแจ้งเตือนการสูบบุหรี่ในพื้นที่ห้ามสูบผ่าน LINE Notify**. การประชุมวิชาการและนำเสนอผลงานวิจัยระดับชาติวิทยาศาสตร์ วิศวกรรมศาสตร์ เกษตรศาสตร์ และเทคโนโลยีครั้งที่ 1.

สาวิตรี วงษ์นุ่น. (2563). การประยุกต์รูปแบบ ICT แจ้งเตือนสารสนเทศด้วยเทคนิค Line notify API ในสถานการณ์ COVID-19. **วารสารวิชาการ ปขมท**. 9(3): 178 - 187.

สันต์ทศน์ สุริยันต์. (2543). **การพัฒนาเครื่องมือสำหรับจัดการกระบวนการวางแผนการทดสอบซอฟต์แวร์**. (วิทยานิพนธ์ ปริญญามหาบัณฑิต). จุฬาลงกรณ์มหาวิทยาลัย.

อิมรอน แวมง อิสมาแอล พิทักษ์ลาวัลย์ วันอัฟฟาน มิมอ และ พรรณี แพงทิพย์. (2563). การพัฒนาเว็บแอปพลิเคชันขอใช้บริการถ่ายสำเนาเอกสารทางวิชาการร่วมกับการแจ้งเตือนผ่านแอปพลิเคชันไลน์: กรณีศึกษาคณะวิทยาศาสตร์เทคโนโลยีและการเกษตร มหาวิทยาลัยราชภัฏยะลา. **วารสารวิชาการการประยุกต์ใช้เทคโนโลยีสารสนเทศ** 6(2):61 – 71.

Chai Phonbopit. (2562). **Deploy เว็บไซต์ React ฟรีๆ ด้วย Firebase Hosting**. ค้นเมื่อ 10 ตุลาคม 2566 ค้นจาก https://blog.devahoy.com/blog/2019/01/how-to-deploy-react-with-firebase-hosting.

Chai Phonbopit. (2565). **ทำระบบ Login/Register ด้วย React Router และ Firebase Auth (JavaScript)**. ค้นเมื่อ 20 กรกฎาคม 2566 ค้นจาก https://devahoy.com/workshop-react-firebase-authentication-javascript.

Prudchaya Palee. (2563). **สร้าง REST API ด้วย Firebase cloud functions และ Express**. ค้นเมื่อ 5 สิงหาคม 2566 ค้นจาก https://prudchayapalee.medium.com/cloud-firestore-firebase-functions-express-js-606ac55a7287.