

บทที่ 2

การศึกษาค้นคว้าและงานที่เกี่ยวข้อง

การพัฒนาโครงงานเรื่อง การทดสอบระบบบริหารจัดการงานสนับสนุนภายในองค์กรด้วยการทดสอบอัตโนมัติ เป็นการทดสอบระบบในระดับ System Testing และเน้นการใช้ Automated เพื่อตรวจสอบความถูกต้องและความเสถียรของระบบ โดยมีการศึกษาหาความรู้และงานที่เกี่ยวข้องสำหรับการพัฒนาการทดสอบระบบในครั้งนี้ มีข้อมูลต่างๆที่ได้นำมาศึกษาค้นคว้าดังนี้

2.1 ระบบงานเดิมของการทดสอบระบบบริหารจัดการงานสนับสนุนภายในองค์กร

ก่อนนำแนวทางการทดสอบแบบอัตโนมัติมาใช้ บริษัท ลูคัส สแตรททิจี จำกัด ดำเนินการทดสอบระบบบริหารจัดการงานสนับสนุนภายในองค์กรในระดับ System Testing โดยใช้วิธีการ Manual Testing เป็นหลัก ซึ่งผู้ทดสอบจะทำงานตามขั้นตอนทดสอบและตรวจสอบผลลัพธ์ตาม Test Case ที่กำหนดอย่างละเอียด แม้ว่าวิธีการนี้จะสามารถตรวจสอบพฤติกรรมของระบบในบางสถานการณ์ได้อย่างตรงจุด แต่ยังคงมีข้อจำกัดสำคัญดังนี้

1. ใช้เวลานานในการดำเนินการทดสอบในแต่ละครั้ง โดยเฉพาะเมื่อจำเป็นต้องทดสอบฟีเจอร์ต่าง ๆ ซ้ำหลายรอบ
2. มีความเสี่ยงต่อความผิดพลาดจากมนุษย์ (Human Error) เช่น การลืมคลิกปุ่ม หรือการตีความผลลัพธ์ผิดพลาด
3. ไม่เหมาะสมกับการทดสอบซ้ำ (Regression Testing) ซึ่งเป็นการทดสอบที่จำเป็นต้องดำเนินการทุกครั้งหลังการปรับปรุงหรือเพิ่มเติมระบบ

ข้อจำกัดเหล่านี้ส่งผลให้กระบวนการพัฒนาและส่งมอบระบบเกิดความล่าช้า รวมถึงลดความน่าเชื่อถือของผลการทดสอบในระยะยาว ดังนั้น จึงมีความจำเป็นที่จะต้องปรับปรุงกระบวนการทดสอบด้วยการนำแนวทางการทดสอบแบบอัตโนมัติมาใช้ เพื่อเพิ่มประสิทธิภาพ ความรวดเร็ว และความแม่นยำ รวมถึงความสามารถในการทดสอบซ้ำได้อย่างสม่ำเสมอ

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 การทดสอบซอฟต์แวร์ (Software Testing) คือการทดสอบระบบซอฟต์แวร์ในระดับสูงสุด เพื่อประเมินว่า ส่วนประกอบต่าง ๆ ของระบบที่ถูกรวมเข้าด้วยกัน สามารถทำงานได้ถูกต้องตามข้อกำหนดหรือไม่ โดยเป็นการทดสอบแบบ Black-box Testing ซึ่งผู้ทดสอบไม่จำเป็นต้องทราบรายละเอียดภายในของโค้ด แต่จะเน้นการตรวจสอบผลลัพธ์ที่แสดงต่อผู้ใช้เป็นหลัก ประเภทของ System Testing ที่พบบ่อย ได้แก่

1. Functional Testing เป็นการตรวจสอบว่าฟังก์ชันต่าง ๆ ของระบบทำงานถูกต้องตามความต้องการ (Requirement) หรือไม่
 2. Usability Testing เป็นการประเมินความง่ายในการใช้งานของระบบ
 3. Performance Testing ใช้ตรวจสอบความเร็วและความเสถียรของระบบภายใต้ภาระการใช้งานที่หลากหลาย
 4. Security Testing เป็นการทดสอบเพื่อค้นหาจุดอ่อนและประเมินความปลอดภัยของระบบ
- ในโครงการนี้ จะมุ่งเน้นการทดสอบแบบ Functional Testing และ Regression Testing เป็นหลัก เพื่อให้มั่นใจว่าการทำงานของระบบยังคงถูกต้องและความเสถียรหลังจากการเปลี่ยนแปลงหรือเพิ่มเติมฟีเจอร์ใหม่

2.2.2 แนวคิด Page Object Model (POM) คือรูปแบบการออกแบบโครงสร้างชุดทดสอบที่แยกส่วนของ “หน้าเว็บ” และ “ตรรกะการทดสอบ” ออกจากกันอย่างชัดเจน ซึ่งช่วยให้สคริปต์ทดสอบมีความเป็นโมดูล (Modular) เพิ่มความสามารถในการดูแลรักษา (Maintainability) และสามารถใช้ซ้ำ (Reusability) ได้ดีขึ้น ในแต่ละหน้าจอของระบบจะถูกกำหนดเป็น “Page Object” ซึ่งประกอบด้วยองค์ประกอบ (Elements) และการกระทำ (Actions) ต่าง ๆ เช่น ปุ่ม กรอกข้อมูล หรือข้อความแสดงผล โดย Page Object จะถูกเรียกใช้งานจาก Test Script โดยตรง ส่งผลให้เมื่อมีการเปลี่ยนแปลงในส่วน of UI สามารถแก้ไขได้ในตำแหน่งเดียวโดยไม่ต้องแก้ไขหลายจุดในสคริปต์ทดสอบ

2.2.3 Accessibility Testing หรือ การทดสอบการเข้าถึงเว็บไซต์ คือกระบวนการตรวจสอบว่าเว็บไซต์หรือแอปพลิเคชันสามารถเข้าถึงและใช้งานได้โดยผู้ใช้ทุกกลุ่ม รวมถึงผู้พิการหรือผู้ที่มีข้อจำกัดทางร่างกายและประสาทสัมผัส ซึ่งเป็นสิ่งสำคัญในการสร้างประสบการณ์ใช้งานที่เท่าเทียมและไม่เป็นอุปสรรคต่อผู้ใช้งานทุกคน ตัวอย่างรูปแบบของการทดสอบ Accessibility ได้แก่

1. การทดสอบด้วยโปรแกรมช่วยอ่านหน้าจอ (Screen Reader Testing) เพื่อประเมินว่าเนื้อหาและฟังก์ชันต่าง ๆ สามารถถูกอ่านและนำทางได้อย่างถูกต้อง
2. การทดสอบคีย์บอร์ด (Keyboard Testing) ตรวจสอบว่าผู้ใช้สามารถนำทางเว็บไซต์หรือแอปพลิเคชันโดยใช้คีย์บอร์ดอย่างครบถ้วนโดยไม่ต้องใช้เมาส์

3. การทดสอบการใช้สีและความคมชัด (Color Contrast Testing) เพื่อให้แน่ใจว่าสีตัวอักษรและพื้นหลังมีความแตกต่างกันเพียงพอสำหรับผู้ที่มีปัญหาด้านการมองเห็น
4. การทดสอบการใช้งานเทคโนโลยีช่วยเหลือ (Assistive Technology Testing) เช่น โปรแกรมขยายหน้าจอ หรืออุปกรณ์ควบคุมด้วยเสียง
5. การทดสอบการแสดงผลบนอุปกรณ์และเบราว์เซอร์หลากหลาย: เพื่อให้แน่ใจว่าเว็บไซต์ทำงานได้ดีในทุกสภาพแวดล้อม

การทดสอบ Accessibility จึงเป็นส่วนสำคัญของกระบวนการพัฒนาซอฟต์แวร์ที่ตอบสนองความต้องการของผู้ใช้ทุกกลุ่ม และสอดคล้องกับมาตรฐานสากล เช่น WCAG (Web Content Accessibility Guidelines) ซึ่งช่วยลดปัญหาและเพิ่มความพึงพอใจในการใช้งานให้กับผู้ใช้ทุกประเภท

2.3 งานวิจัย Page Object Model (POM) และ Accessibility Testing

2.3.1 Leotta, M., Stocco, A., & Ricca, F. (2015). Improving test suites maintainability with the Page Object pattern: An industrial case study. งานวิจัยนี้ศึกษาการนำ POM มาใช้ในสภาพแวดล้อมการพัฒนาซอฟต์แวร์จริงขององค์กรหนึ่ง โดยวิเคราะห์ผลกระทบของการใช้ POM ต่อความง่ายในการดูแลรักษาชุดทดสอบและความสามารถในการปรับตัวต่อการเปลี่ยนแปลงของระบบ พบว่า POM ช่วยลดเวลาที่ต้องใช้ในการแก้ไขสคริปต์ทดสอบหลังการเปลี่ยนแปลง UI ได้มากกว่าวิธีดั้งเดิมอย่างมีนัยสำคัญ และช่วยเพิ่มความชัดเจนของโค้ดและประสิทธิภาพการทำงานร่วมกันในทีม

2.3.2 Stocco, A., Leotta, M., & Ricca, F. (2017). APOGEN: Automatic page object generator for web testing. บทความนำเสนอ APOGEN เครื่องมือสร้าง POM อัตโนมัติด้วยการวิเคราะห์ DOM ของเว็บเพจ ช่วยลดงานเขียนโค้ดซ้ำซ้อนและข้อผิดพลาดจากมนุษย์ ผลการประเมินพบว่า APOGEN สร้างโครงสร้าง POM ถูกต้อง ใช้งานได้จริง และช่วยเพิ่มความรวดเร็วและความถูกต้องในการสร้างชุดทดสอบ

2.3.3 Chen, Y., Li, Z., Zhao, R., & Guo, J. (2019). Research on Page Object generation approach for web application testing. เสนอวิธีสร้าง Page Object อัตโนมัติด้วยเทคนิคจับคู่ pattern และ machine learning เพื่อช่วยลดงานเขียนโค้ดด้วยมือ และเพิ่มคุณภาพชุดทดสอบ ระบบวิเคราะห์ DOM และจัดกลุ่มองค์ประกอบอย่างมีประสิทธิภาพ ผลทดลองกับเว็บแอปจริงแสดงว่าวิธีนี้ช่วยลดเวลาพัฒนาและปรับปรุงคุณภาพชุดทดสอบ

2.3.4 Neelapu, M. (2024). Enhancing test automation with an advanced Page Object Model for scalable and maintainable web applications. เสนอการพัฒนาต่อยอด POM โดยใช้ modular architecture และการจัดการ locator แบบมีระบบ รวมถึงจัดการ dependency ระหว่าง Page Objects เพื่อให้ชุดทดสอบขยายตัวง่าย ลดโค้ดซ้ำซ้อน รองรับ parallel testing และลดข้อผิดพลาดในการจัดการ UI

2.3.5 Ganioglu, C. (2025). Understanding the significance of Page Object Model (POM) in software test frameworks. เน้นความรู้และความสำคัญของ POM ในการพัฒนาระบบทดสอบอัตโนมัติ อธิบายจุดเด่นที่ช่วยแยกตรรกะ UI ออกจากชุดทดสอบ ทำให้โค้ดเป็นระเบียบและง่ายต่อการบำรุงรักษา พร้อมกรณีศึกษาและตัวอย่างการใช้ POM กับ Selenium และ Playwright

2.3.6 Herath, A. (2024). POM under the hood: Technical deep dive into the Page Object Model. เจาะลึกโครงสร้างทางเทคนิคของ POM รวมถึงการออกแบบ Page Object การจัดการ locator แบบต่าง ๆ และกลยุทธ์รับมือกับ UI ที่เปลี่ยนแปลงบ่อย เช่น dynamic locators และ pattern-based selectors พร้อมตัวอย่างการใช้ POM ในเฟรมเวิร์กต่าง ๆ

2.3.7 Sánchez-Gordón, M. L., & Luján-Mora, S. (2017). A method for accessibility testing of web applications in agile environments. เสนอวิธีทดสอบ accessibility ของเว็บแอปในสภาพแวดล้อม Agile ด้วยการผสมผสานเครื่องมืออัตโนมัติที่สอดคล้อง WCAG การทดสอบผู้เชี่ยวชาญ และ user testing เพื่อครอบคลุมทุกมิติของ accessibility ช่วยให้ตรวจจับปัญหาและปรับปรุงเว็บได้รวดเร็ว

2.3.8 Sane, P. (2021). A brief survey of current software engineering practices in continuous integration and automated accessibility testing. ทบทวนการบูรณาการทดสอบ accessibility อัตโนมัติในกระบวนการ CI/CD รวมถึงเครื่องมือยอดนิยม เช่น Axe, Lighthouse และ Pa11y เน้นความท้าทายและข้อจำกัดในการบูรณาการ และความสำคัญของการสร้าง culture ด้าน accessibility ในทีมพัฒนา

2.3.9 Ara, J., & Sánchez-Gordón, M. L. (2023). Accessibility engineering in web evaluation process: A systematic literature review. ทบทวนวรรณกรรมเชิงระบบเกี่ยวกับวิศวกรรม accessibility ในการประเมินเว็บระหว่างปี 2010-2021 วิเคราะห์แนวทางต่าง ๆ เช่น เครื่องมืออัตโนมัติ การประเมินด้วยผู้ใช้ และมาตรฐาน WCAG ชี้ช่องว่างในการบูรณาการ accessibility testing เข้ากับวงจรพัฒนาซอฟต์แวร์

2.3.10 Ara, J., & Sánchez-Gordón, M. L. (2024). An inclusive framework for automated web content accessibility evaluation. นำเสนอกรอบงาน multidimensional สำหรับประเมิน accessibility ของเนื้อหาเว็บแบบอัตโนมัติ แก้อัปเดตของเครื่องมือปัจจุบันโดยวิเคราะห์ DOM, ARIA, การใช้งานคีย์บอร์ด และ assistive technologies ลด false positives และเพิ่มความแม่นยำ กรอบงานถูกประเมินกับเว็บตัวอย่างหลากหลายและแสดงประสิทธิภาพเหนือกว่าเครื่องมือทั่วไป

2.4 งานวิจัยและโครงการที่เกี่ยวข้อง

2.4.1 เอกสารทางการของ Playwright เป็นสำหรับการทดสอบอัตโนมัติแบบ End-to-End ที่ได้รับความนิยมในปัจจุบัน โดยเฉพาะในการทดสอบเว็บแอปพลิเคชันผ่านเบราว์เซอร์หลากหลายชนิด ซึ่งให้ข้อมูลเกี่ยวกับการติดตั้ง การเริ่มต้นใช้งาน และตัวอย่างโค้ดเบื้องต้นสำหรับการเขียนสคริปต์ทดสอบแบบอัตโนมัติ ในโครงการนี้ได้นำแนวทางจากเอกสารดังกล่าวมาใช้ในการออกแบบและเขียนสคริปต์ทดสอบด้วยแนวคิด Page Object Model (POM) ซึ่งช่วยให้โครงสร้างสคริปต์ทดสอบมีความชัดเจนและดูแลรักษาได้ง่าย

2.4.2 โครงการระบบจัดการการแจ้งเตือนผ่านไลน์ (Notification Management via Line) เป็นโครงการของนักศึกษารุ่นปี 2563 ที่พัฒนาระบบสำหรับจัดการการแจ้งเตือนผ่านช่องทาง LINE ซึ่งเป็นแอปพลิเคชันส่งข้อความยอดนิยมในประเทศไทย ระบบดังกล่าวมีวัตถุประสงค์เพื่อช่วยให้การสื่อสารข้อมูลแจ้งเตือนต่าง ๆ เป็นไปอย่างรวดเร็วและมีประสิทธิภาพ เช่น การแจ้งเตือนข่าวสารสำคัญ การแจ้งเตือนงาน หรือการแจ้งเตือนเหตุการณ์ต่าง ๆ ภายในองค์กร โดยเน้นความสะดวกในการตั้งค่าการแจ้งเตือนและการบริหารจัดการผู้รับสารผ่าน LINE Official Account ซึ่งเหมาะสมกับการใช้งานทั้งในองค์กรและธุรกิจขนาดต่าง ๆ

2.4.3 บทความวิจัย ระบบรายงานปัญหาการใช้งานโปรแกรมของลูกค้าและการแจ้งเตือนผ่านไลน์ บทความวิจัยชิ้นนี้ได้รับการตีพิมพ์ในวารสารแม่โจ้เทคโนโลยีสารสนเทศและนวัตกรรม (Maejo Information Technology and Innovation Journal – MITIJ) และวารสารแม่โจ้เทคโนโลยีสารสนเทศ (Maejo Information Technology Journal – MITJ) โดยเน้นศึกษาระบบที่ช่วยให้ลูกค้าสามารถรายงานปัญหาที่พบเจอระหว่างการใช้งานโปรแกรมได้อย่างง่ายดาย ผ่านช่องทางดิจิทัล รวมถึงระบบแจ้งเตือนผ่าน LINE ที่ช่วยส่งข้อมูลปัญหาไปยังเจ้าหน้าที่ที่รับผิดชอบโดยทันที ทำให้กระบวนการแก้ไขปัญหาเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ บทความดังกล่าวยังกล่าวถึงประโยชน์ของการเชื่อมต่อบริการรายงานปัญหากับแพลตฟอร์มการแจ้งเตือนผ่าน LINE ซึ่งช่วยเพิ่มความสะดวกและการตอบสนองต่อปัญหาในเวลาจริง (Real-time Notification)

2.4.4 งานวิจัยของประทีป เทพยศ และอภิรณธ์ อังสุรัตน์ (2564) ได้กล่าวถึงการพัฒนาระบบแจ้งเตือนออนไลน์ที่ช่วยให้ผู้ใช้สามารถแจ้งซ่อมและตรวจสอบสถานะได้อย่างสะดวก รวมถึงการใช้ระบบจัดเก็บข้อมูลบนฐานข้อมูลคลาวด์ (Cloud Database) ซึ่งเพิ่มความสะดวกในการเข้าถึงและจัดการข้อมูลของผู้ใช้จากทุกที่ งานวิจัยดังกล่าวมีความสำคัญในการนำแนวคิดมาใช้พัฒนาระบบให้มีความเป็นระบบมากขึ้น และสามารถตอบสนองความต้องการของผู้ใช้งานได้อย่างมีประสิทธิภาพ