

WARERING: นวัตกรรมแพลตฟอร์มไอโอทีเพื่อการควบคุมและแสดงผลแบบเรียลไทม์ ในบริบทการเรียนรู้

WARERING: AN INNOVATIVE IOT PLATFORM FOR REAL-TIME CONTROL AND VISUALIZATION IN LEARNING CONTEXTS

สุพิชฌาย์ จันทร์เรือง¹⁾, ศักรินทร์ สิงห์อยู่²⁾, กัตพัทธ์ สมหวัง³⁾, ณาตทชา มุมแดง⁴⁾, สุรพงศ์ แก้วนิล⁵⁾, พิชยา สุขปลั่ง^{6)*}
และ ศุภกฤษ นาคป้อมฉิน⁷⁾

¹⁾ อาจารย์ประจำสาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม

²⁾⁻⁵⁾ นักศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม

⁶⁾⁻⁷⁾ อาจารย์ประจำสาขาวิชาเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม

Suphitcha Chanrueang^{1)*}, Sakarin Singhayu²⁾, Gatpat Somwang³⁾, Natthacha Mumdaeng⁴⁾,

Surapong Kaewnin⁵⁾, Pichaya Sookplung^{6)*} and Supakit Nakpomchin⁷⁾

¹⁾ Lecturer of Software Engineering Department, Faculty of Science and Technology,
Nakhon Pathom Rajabhat University

²⁾⁻⁵⁾ Bachelor's degree student of Software Engineering Department, Faculty of Science
and Technology, Nakhon Pathom Rajabhat University

⁶⁾⁻⁷⁾ Lecturer of Information Technology Department, Faculty of Science and Technology,
Nakhon Pathom Rajabhat University

*Corresponding author e-mail: pichaya@webmail.npru.ac.th

Receive: May 09, 2025

Revised: June 10, 2025

Accepted: July 31, 2025

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาและประเมินแพลตฟอร์มต้นแบบด้านอินเทอร์เน็ตของสรรพสิ่ง (IoT) สำหรับใช้ในการเรียนการสอนเชิงปฏิบัติในระดับอุดมศึกษา โดยมุ่งเน้นการออกแบบให้ใช้งานง่าย รองรับผู้ใช้จำนวนมาก และสามารถควบคุมอุปกรณ์และแสดงผลข้อมูลจากเซ็นเซอร์ได้แบบเวลาจริง การดำเนินงานเริ่มจากการวิเคราะห์ข้อกำหนดของแพลตฟอร์มที่มีใช้อยู่ในปัจจุบัน ซึ่งยังไม่เหมาะสมกับสภาพแวดล้อมของห้องเรียน ทั้งในด้านความยุ่งยากในการใช้งานและข้อกำหนดในการควบคุมอุปกรณ์พร้อมกันหลายชุด คณะผู้วิจัยจึงพัฒนาแพลตฟอร์ม “WARERING” โดยประยุกต์ใช้แนวคิดการพัฒนาซอฟต์แวร์แบบอไจล์ (Agile Software Development) ภายใต้อกรอบการทำงานแบบสกรัม (Scrum) ซึ่งเปิดโอกาสให้พัฒนาแบบวนรอบและปรับปรุงอย่างต่อเนื่องตามข้อเสนอแนะของผู้ใช้งาน เทคโนโลยีหลักที่นำมาใช้ ได้แก่ โพรโตคอล MQTT สำหรับการรับ-ส่งข้อมูล WebSocket สำหรับการแสดงผลแบบเวลาจริง MongoDB สำหรับการจัดเก็บข้อมูลแบบไม่มีโครงสร้าง และ Kubernetes สำหรับการจัดการระบบบนบริการคลาวด์ ผลการทดสอบระบบโดยจำลองการใช้งานจริง พบว่าแพลตฟอร์มสามารถรองรับผู้ใช้งานพร้อมกันได้ถึง 500 ราย โดยมีค่าเวลาตอบสนองเฉลี่ยไม่เกิน 500 มิลลิวินาที และอัตราความผิดพลาดไม่เกินร้อยละ 1 ส่วนผลการประเมินจากผู้ใช้งานจริง พบว่า ผู้ใช้งานมีความพึงพอใจต่อระบบอยู่ในระดับมากที่สุด โดยมีค่าเฉลี่ยเท่ากับ 4.45 ค่าส่วนเบี่ยงเบนมาตรฐานเท่ากับ 0.53 โดยเฉพาะด้านความรวดเร็ว

ในการเรียนรู้ระบบผู้ที่มีความพึงพอใจอยู่ในระดับมากที่สุด โดยมีค่าเฉลี่ยเท่ากับ 4.63 ค่าส่วนเบี่ยงเบนมาตรฐานเท่ากับ 0.34 ผลลัพธ์เหล่านี้แสดงให้เห็นว่าแพลตฟอร์ม WAREING มีศักยภาพในการรองรับการเรียนรู้เชิงปฏิบัติแบบเรียลไทม์ได้อย่างมีประสิทธิภาพ และสามารถนำไปประยุกต์ใช้ในสถาบันการศึกษาอื่นได้ในอนาคต

คำสำคัญ: อินเทอร์เน็ตของสรรพสิ่ง (ไอโอที) แพลตฟอร์มการเรียนรู้ การพัฒนาซอฟต์แวร์แบบโอเจส การควบคุมแบบเรียลไทม์ การแสดงผลข้อมูล

ABSTRACT

This research aimed to develop and evaluate a prototype Internet of Things (IoT) platform for practical instruction in higher education. The design focused on ease of use, scalability for large numbers of users, and the ability to control devices and visualize sensor data in real time. The study began with an analysis of the limitations of existing platforms, which were found to be unsuitable for classroom environments due to their complexity and limited capacity to manage multiple devices concurrently. To address these issues, the research team developed “WAREING,” applying Agile Software Development principles within the Scrum framework to enable iterative development and continuous improvement based on user feedback. Key technologies integrated into the platform include MQTT for data transmission, WebSocket for real-time visualization, MongoDB for unstructured data storage, and Kubernetes for cloud-based system management. Performance testing through simulated real-world usage demonstrated that the platform could support up to 500 concurrent users with an average response time of less than 500 milliseconds and an error rate of less than 1%. User satisfaction assessments indicated a very high level of satisfaction, with a mean score of 4.45 and a standard deviation of 0.53. Notably, the aspect of ease of learning the system received the highest satisfaction rating, with a mean score of 4.63 and a standard deviation of 0.34. These findings suggest that the WAREING platform effectively supports real-time, hands-on learning and demonstrates strong potential for broader adoption across educational institutions.

Keywords: Internet of Things (IoT), Learning Platform, Agile Software Development, Real-Time Control, Data Visualization

บทนำ

ในปัจจุบัน เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things: IoT) ได้กลายเป็นเครื่องมือสำคัญที่ช่วยสนับสนุนการเรียนรู้เชิงประสบการณ์ โดยเฉพาะในระดับอุดมศึกษา ซึ่งมีกิจกรรมการเรียนการสอนที่เน้น

การปฏิบัติจริง เช่น การพัฒนาระบบฝังตัว การควบคุมอุปกรณ์ระยะไกล และการแสดงผลข้อมูลจากเซนเซอร์แบบเวลาจริง เทคโนโลยีเหล่านี้ ช่วยส่งเสริมการเรียนรู้ผ่านการลงมือปฏิบัติ การรับรู้ผลลัพธ์ทันที และการจำลองสถานการณ์ที่ใกล้เคียงกับโลกแห่งความจริง ซึ่งเอื้อต่อ

การพัฒนาทักษะเชิงเทคนิคและการคิดเชิงระบบของผู้เรียน (Hwang et al., 2022, 4725) อย่างไรก็ตาม การนำแพลตฟอร์ม IoT มาใช้ในสถานศึกษายังคงประสบปัญหาหลายด้าน ทั้งในแง่ข้อจำกัดทางเทคนิค ความซับซ้อนในการติดตั้ง และความไม่เหมาะสมกับบริบทการเรียนรู้ของผู้ใช้ในชั้นเรียน (Hong et al., 2023, 6)

แพลตฟอร์ม IoT ที่นิยมใช้งานในปัจจุบัน เช่น Node-RED, ThingsBoard และ Blynk ต่างมีข้อจำกัดสำคัญเมื่อนำมาใช้งานในบริบทการเรียนการสอน Node-RED แม้จะมีความยืดหยุ่นสูงแต่ต้องอาศัยทักษะการเขียนโค้ดขั้นสูง (Ghashim and Arshad, 2023, 13) ThingsBoard มีที่เจอร์ครบถ้วนแต่ขั้นตอนการติดตั้งและจัดการสิทธิ์ผู้ใช้ซับซ้อน ส่วน Blynk ใช้งานง่ายแต่มีข้อจำกัดในด้านการจัดการผู้ใช้ การปรับแต่งส่วนของแดชบอร์ด และการปรับขยายระบบ (Boca et al., 2023, 13) เพื่อแก้ไขข้อจำกัดเหล่านี้ คณะผู้วิจัยจึงพัฒนาแพลตฟอร์มต้นแบบ “WARERING” เพื่อใช้ในการเรียนรู้เชิงปฏิบัติสำหรับนักศึกษาในระดับอุดมศึกษา โดยเน้นความง่ายในการติดตั้ง การใช้งานที่ไม่ยุ่งยากซับซ้อน ความสามารถในการรองรับผู้ใช้จำนวนมาก และการแสดงผลข้อมูลแบบเรียลไทม์ โดย WARERING ใช้ WebSocket ในการแสดงผล และโปรโตคอล MQTT สำหรับควบคุมอุปกรณ์ทั้งหมดที่เชื่อมต่อ ซึ่งเหมาะสมกับเครือข่ายที่มีข้อจำกัดด้านพลังงานและแบนด์วิดท์ (ธงชัย และคณะ, 2566, 112; Hwang et al., 2022, 4727) โดยแพลตฟอร์มมีการใช้ Redis เพื่อเพิ่มประสิทธิภาพในการเข้าถึงข้อมูล และใช้ MongoDB สำหรับจัดเก็บข้อมูลที่ได้รับจากอุปกรณ์ (Mdemaya et al., 2024, 1206)

แพลตฟอร์ม WARERING ถูกออกแบบให้สามารถติดตั้งบนบริการคลาวด์ เช่น Amazon EC2 และรองรับการปรับขยายระบบผ่าน Kubernetes เพื่อให้สามารถบริหารจัดการทรัพยากรได้อย่างมีประสิทธิภาพ รองรับจำนวนผู้ใช้ได้อย่างยืดหยุ่น (Böhm and Wirtz, 2022,

5; Pahl et al., 2018, 7; Bhaskaran et al., 2024, 11) ระบบนี้ยังคำนึงถึงความง่ายในการเริ่มต้นใช้งานสำหรับผู้ที่ไม่มีความรู้พื้นฐานด้านเครือข่ายหรือ DevOps ซึ่งช่วยลดอุปสรรคในการนำไปใช้ในสถานศึกษา ในด้านการพัฒนาระบบนี้ยึดแนวทางการพัฒนาซอฟต์แวร์แบบอไจล์ ภายใต้กรอบการทำงานแบบสกรัม ที่เน้นการพัฒนาเป็นรอบสั้น ๆ รับฟังข้อเสนอแนะจากผู้ใช้งานจริง และปรับปรุงระบบอย่างต่อเนื่อง (Beck et al., 2001) สำหรับการประเมินประสิทธิภาพ ได้ใช้เครื่องมือมาตรฐาน ได้แก่ Apache JMeter สำหรับทดสอบภาระงาน (Han, 2021, 480) Wireshark สำหรับวิเคราะห์การสื่อสารในระดับเครือข่าย และเครื่องมืออัตโนมัติอย่าง Cypress และ Selenium สำหรับทดสอบส่วนติดต่อผู้ใช้ (Bhimanapati et al., 2024, 72) ซึ่งสอดคล้องกับแนวทางการพัฒนาแบบต่อเนื่ององค์ความรู้จากวรรณกรรมและงานวิจัยที่เกี่ยวข้อง ได้ถูกนำมาประยุกต์ใช้อย่างเหมาะสมตลอดกระบวนการออกแบบ พัฒนา และการทดสอบระบบ WARERING เพื่อให้ตอบสนองความต้องการของการเรียนรู้เชิงปฏิบัติในระดับอุดมศึกษาได้อย่างมีประสิทธิภาพ ยืดหยุ่น และสามารถต่อยอดหรือขยายผลในบริบทอื่นได้อย่างยั่งยืน

วัตถุประสงค์ของการวิจัย

1. เพื่อศึกษาข้อจำกัดของแพลตฟอร์ม IoT แบบเรียลไทม์ที่ใช้อยู่ในปัจจุบันและวิเคราะห์ช่องว่างขององค์ความรู้ในบริบทการเรียนรู้ระดับอุดมศึกษา
2. เพื่อออกแบบและพัฒนาแพลตฟอร์ม WARERING เพื่อรองรับการควบคุมอุปกรณ์ การแสดงผลข้อมูลแบบเวลาจริง และการใช้งานที่ยืดหยุ่น โดยใช้แนวทางการพัฒนาซอฟต์แวร์แบบอไจล์
3. เพื่อประเมินประสิทธิภาพของระบบผ่านการทดสอบด้านภาระงาน ความเร็ว และความเสถียร โดยใช้เครื่องมือ JMeter และ Wireshark ร่วมกับเกณฑ์มาตรฐานทางเทคนิคที่เกี่ยวข้อง

4. เพื่อศึกษาระดับความพึงพอใจของผู้ใช้งานต่อระบบ โดยใช้แบบสอบถามเป็นเครื่องมือในการเก็บรวบรวมข้อมูล

ขอบเขตของการวิจัย

งานวิจัยนี้พัฒนาแพลตฟอร์มไอโอทีต้นแบบสำหรับควบคุมและแสดงข้อมูลจากอุปกรณ์ตรวจวัดแบบเรียลไทม์ในบริบทการเรียนรู้ขอบเขตดังนี้

1. ระบบรองรับการเชื่อมต่ออุปกรณ์ไอโอทีโดยโพรโทคอล MQTT

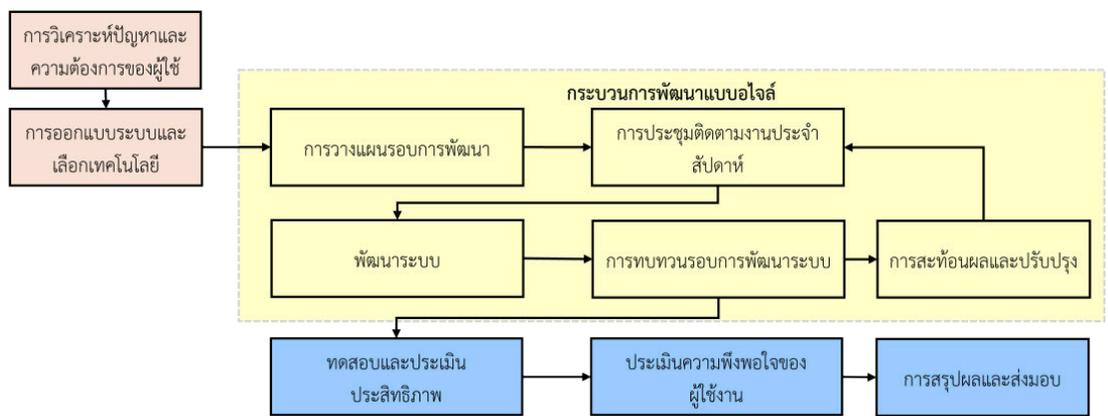
2. ผู้ใช้งานสามารถลงทะเบียน เข้าสู่ระบบ และจัดการบัญชีผู้ใช้ของตนเองได้

3. ผู้ใช้สามารถควบคุมอุปกรณ์ไอโอทีและดูข้อมูลจากเซ็นเซอร์ในรูปแบบ JSON ได้แบบเวลาจริง

4. ผู้ใช้สามารถเพิ่ม ลบ และจัดการส่วนแสดงผลและปุ่มควบคุมในแดชบอร์ดได้ตามต้องการ

5. ระบบรองรับการจัดเก็บข้อมูลลงในฐานข้อมูล MongoDB และส่งออกข้อมูลเป็นไฟล์ JSON หรือ Excel

6. ผู้ดูแลระบบสามารถจัดการข้อมูลผู้ใช้งานและโพรโทคอล MQTT ผ่านระบบหลังบ้าน



ภาพที่ 1 กรอบแนวคิดงานวิจัย

วิธีดำเนินการวิจัย

ผู้วิจัยใช้กระบวนการพัฒนาระบบแบบอเจิล โดยเริ่มจากการวิเคราะห์ปัญหาและความต้องการที่ชัดเจนของผู้ใช้ จากนั้นออกแบบระบบและเลือกเทคโนโลยีที่เหมาะสม ก่อนเข้าสู่กระบวนการพัฒนาแบบวนรอบ ซึ่งประกอบด้วย การวางแผน การประชุมติดตามผล การพัฒนา การทบทวน และการสะท้อนผลเพื่อปรับปรุงในรอบถัดไป หลังพัฒนาแล้วเสร็จ ได้ดำเนินการทดสอบและประเมินประสิทธิภาพระบบ ประเมินความพึงพอใจของผู้ใช้งาน และสรุปผลเพื่อนำไปสู่การส่งมอบระบบที่สมบูรณ์ตามที่แสดงในภาพที่ 1

1. การวิเคราะห์ปัญหาและความต้องการของแพลตฟอร์มอินเทอร์เน็ตของสรรพสิ่งที่มีอยู่ในปัจจุบัน เช่น Node-RED, ThingsBoard และ Blynk คณะผู้วิจัยได้ดำเนินการศึกษาโดยเริ่มจากการทบทวนวรรณกรรมที่เกี่ยวข้องเพื่อรวบรวมข้อจำกัดและจุดอ่อนของแพลตฟอร์มเหล่านี้ในแง่ของการใช้งานในบริบททางการศึกษา จากนั้นจึงดำเนินการเก็บข้อมูลภาคสนามผ่านการสัมภาษณ์แบบกึ่งโครงสร้างกับอาจารย์และนักศึกษาศาสนาวิชาเทคโนโลยีสารสนเทศ สาขาวิชาเทคโนโลยีคอมพิวเตอร์ จำนวน 52 คน รวมถึงการสังเกตการณ์การใช้งานจริงในชั้นเรียน โดยเน้นประเด็นด้านความยากในการใช้งาน ความซับซ้อน

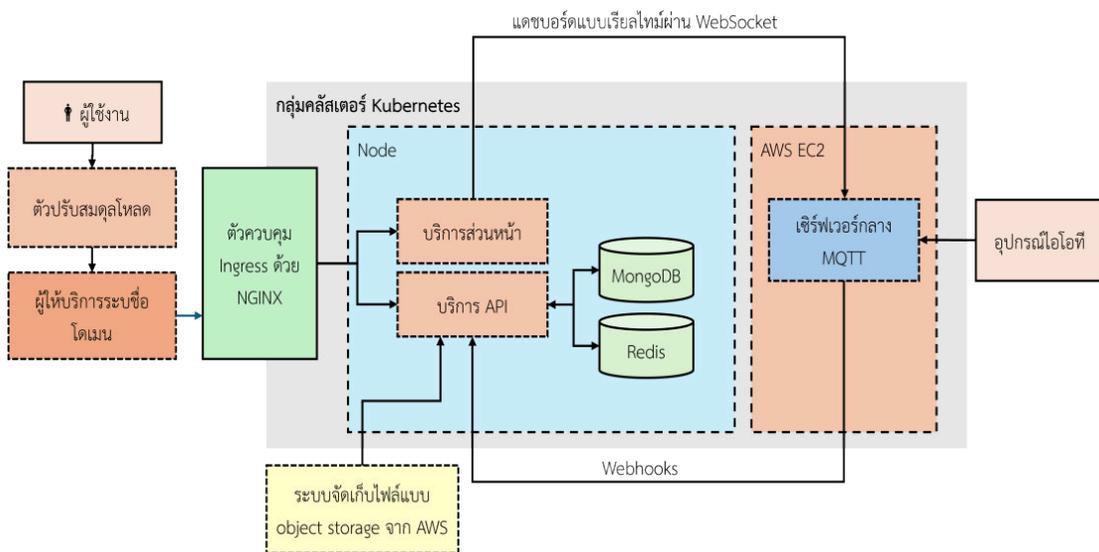
ของการตั้งค่า และความไม่ยืดหยุ่นในการแสดงผลข้อมูล และควบคุมอุปกรณ์แบบเวลาจริง จากข้อมูลที่ได้ คณะผู้วิจัยจึงได้นำมาวิเคราะห์เชิงเปรียบเทียบ เพื่อประเมินคุณสมบัติของแต่ละแพลตฟอร์มและใช้วิธีการวิเคราะห์ช่องว่าง (Gap Analysis) เพื่อระบุความต้องการที่ยังไม่ได้รับการตอบสนองอย่างเหมาะสม ในบริบทของสถานศึกษา ซึ่งผลลัพธ์จากกระบวนการวิเคราะห์ดังกล่าวได้นำไปสู่การกำหนดคุณลักษณะที่จำเป็นของระบบต้นแบบที่ควรได้รับการพัฒนาต่อไป

2. การออกแบบระบบและเลือกเทคโนโลยี หลังจากได้ข้อกำหนดความต้องการของระบบแล้ว คณะผู้วิจัยได้ออกแบบสถาปัตยกรรมของแพลตฟอร์ม WARERING โดยจัดทำแบบจำลองเชิงระบบ ได้แก่ แผนผังสถาปัตยกรรมระบบ แผนผังฐานข้อมูล เพื่อให้เห็นภาพรวมการทำงานของระบบอย่างครบถ้วน แสดงใน

ภาพที่ 2 ในขั้นตอนการเลือกเทคโนโลยี มีการคัดเลือกทั้งฮาร์ดแวร์และซอฟต์แวร์ที่เหมาะสม โดยมีรายละเอียดดังนี้

2.1 การจัดการและประมวลผลข้อมูลใช้บริการคลาวด์ AWS EC2 สำหรับติดตั้งเซิร์ฟเวอร์กลางที่ทำหน้าที่รองรับและประสานงานระบบโดยรวม ข้อมูลจากอุปกรณ์ IoT จะถูกส่งและรับผ่าน EMQX ซึ่งทำหน้าที่เป็น MQTT Broker ในการจัดการการสื่อสารแบบ Publish-Subscribe ทำให้สามารถรับส่งข้อมูลได้แบบเรียลไทม์ มีประสิทธิภาพ และรองรับการเชื่อมต่อจากอุปกรณ์จำนวนมากพร้อมกัน

2.2 การจัดเก็บข้อมูล ใช้ฐานข้อมูลแบบ NoSQL ได้แก่ MongoDB สำหรับข้อมูลเซ็นเซอร์แบบเรียลไทม์ และ Redis สำหรับการจัดการแคชข้อมูลที่ต้องการเข้าถึงรวดเร็ว



ภาพที่ 2 สถาปัตยกรรมของแพลตฟอร์ม WARERING

2.3 การพัฒนาเว็บแอปพลิเคชันประกอบด้วย 2 ส่วนหลัก ได้แก่ ฝั่งเซิร์ฟเวอร์ (Backend) และฝั่งผู้ใช้ (Frontend) โดยฝั่งเซิร์ฟเวอร์พัฒนาด้วยภาษา JavaScript บนสภาพแวดล้อม Node.js เพื่อให้รองรับการประมวลผลแบบไม่ประสานเวลา (Asynchronous)

และมีความยืดหยุ่นในการจัดการกับการเชื่อมต่อของอุปกรณ์จำนวนมาก พร้อมใช้ Express.js ซึ่งเป็นเฟรมเวิร์กที่มีความเบาและทำงานร่วมกับ Node.js ได้อย่างมีประสิทธิภาพ ช่วยจัดการเส้นทาง (Routing) การร้องขอ (Request) และการตอบสนอง (Response)

ของเซิร์ฟเวอร์อย่างเป็นระบบ ส่วนฝั่งผู้พัฒนาด้วย React.js ซึ่งเป็นไลบรารีสำหรับสร้างอินเทอร์เฟซผู้ใช้งาน (User Interface) แบบเชิงองค์ประกอบ (Component-based) ที่สามารถแสดงข้อมูลแบบอินเทอร์แอคทีฟได้อย่างมีประสิทธิภาพ โดยออกแบบแดชบอร์ดให้สามารถแสดงข้อมูลจากเซนเซอร์แบบเรียลไทม์พร้อมทั้งมีส่วนควบคุมการทำงานของอุปกรณ์ในรูปแบบที่ใช้งานง่าย และสามารถตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างมีประสิทธิภาพ

2.4 ระบบรองรับการเชื่อมต่อแบบเรียลไทม์ระหว่างอุปกรณ์กับแพลตฟอร์ม โดยใช้โปรโตคอล MQTT ซึ่งเป็นโปรโตคอลที่มีขนาดเล็ก เหมาะสำหรับระบบ IoT ที่มีข้อจำกัดด้านพลังงานและแบนด์วิดท์ โดยอุปกรณ์ปลายทางจะทำหน้าที่เป็นผู้เผยแพร่ข้อมูล (Publisher) และเซิร์ฟเวอร์ทำหน้าที่เป็นผู้รับข้อมูล (Subscriber) ผ่านตัวกลางที่เรียกว่า MQTT Broker เช่น ซึ่งช่วยให้สามารถส่งข้อมูลเซนเซอร์จากอุปกรณ์มายังเซิร์ฟเวอร์ได้อย่างมีประสิทธิภาพ นอกจากนี้ยังใช้เทคโนโลยี WebSocket สำหรับการสื่อสารแบบสองทางระหว่างเซิร์ฟเวอร์กับเว็บแอปพลิเคชันของผู้ใช้ ทำให้สามารถส่งข้อมูลจากเซิร์ฟเวอร์ไปยังแดชบอร์ดของผู้ใช้งานได้ในทันทีโดยไม่ต้องรีเฟรชหน้าเว็บ จึงช่วยให้การแสดงผลข้อมูลแบบเรียลไทม์ เป็นไปอย่างรวดเร็วและตอบสนองต่อสถานการณ์ปัจจุบันได้อย่างทันทั่วทั้ง

2.5 เครื่องมือที่ใช้ในการพัฒนาและทดสอบระบบในการพัฒนาและตรวจสอบความถูกต้องของระบบได้เลือกใช้เครื่องมือเฉพาะทางเพื่อให้ครอบคลุมทั้งด้านการทำงาน ความเสถียรและประสิทธิภาพ โดยใช้ Cypress สำหรับการทดสอบแบบอัตโนมัติในระดับ End-to-End เพื่อจำลองการใช้งานของผู้ใช้และตรวจสอบความถูกต้องของฟังก์ชันบนอินเทอร์เฟซ Apache JMeter สำหรับการทดสอบภาระงาน (Load Testing) โดยจำลองผู้ใช้งานจำนวนมากเพื่อประเมินประสิทธิภาพและความสามารถในการรองรับโหลดของระบบ และ Wireshark สำหรับการวิเคราะห์ข้อมูลเครือข่าย

ในระดับแพ็คเกจ เพื่อวินิจฉัยปัญหาด้านการสื่อสาร เช่น ความล่าช้า การสูญหายของข้อมูล หรือปัญหาความปลอดภัยการใช้เครื่องมือเหล่านี้ช่วยให้การพัฒนาและทดสอบระบบมีความครอบคลุมและน่าเชื่อถือยิ่งขึ้น

3. การพัฒนาแพลตฟอร์ม WARERING ดำเนินการภายใต้แนวทางการพัฒนาซอฟต์แวร์แบบอจีล์ โดยใช้กรอบการทำงานแบบสกรัม (Scrum Framework) ซึ่งเน้นการพัฒนาแบบวนรอบ (Iterative Development) และการปรับปรุงอย่างต่อเนื่องตามผลสะท้อนจากผู้ใช้งานจริง กระบวนการเริ่มจากการวางแผนในแต่ละรอบการพัฒนา (Sprint Planning) เพื่อกำหนดเป้าหมายฟังก์ชันหลัก และระยะเวลาดำเนินงาน จากนั้นเข้าสู่ขั้นตอนการพัฒนา โดยมีการจัดประชุมเพื่อติดตามความก้าวหน้าและปัญหาที่พบบ่อยอย่างสม่ำเสมอ ในการดำเนินงานแต่ละรอบสิ้นสุดด้วยการนำเสนอผลการพัฒนา (Sprint Review) เพื่อให้ผู้มีส่วนได้ส่วนเสีย ร่วมประเมินความเหมาะสมของระบบ และตามด้วยการประเมินกระบวนการทำงานภายในทีม (Sprint Retrospective) เพื่อวิเคราะห์ประสิทธิภาพการทำงานและวางแผนปรับปรุงในรอบถัดไป กระบวนการพัฒนาระบบในลักษณะนี้เอื้อต่อการตอบสนองต่อความต้องการที่เปลี่ยนแปลงได้อย่างยืดหยุ่น เพิ่มความต่อเนื่องในการมีส่วนร่วมของผู้ใช้งาน และส่งผลต่อคุณภาพของซอฟต์แวร์ที่สามารถใช้งานได้จริงในบริบททางการศึกษา

4. ภายหลังจากการพัฒนาในแต่ละส่วนของระบบเสร็จสิ้น ได้มีการดำเนินการทดสอบการทำงานและประสิทธิภาพของแพลตฟอร์ม IoT ที่พัฒนาขึ้นอย่างครอบคลุม โดยแบ่งออกเป็นการทดสอบเชิงฟังก์ชัน (Functional Testing) และการทดสอบเชิงประสิทธิภาพ (Performance Testing) สำหรับการทดสอบเชิงฟังก์ชันระบบได้รับการตรวจสอบอย่างเป็นระบบโดยใช้ชุดกรณีทดสอบ (Test Cases) เพื่อประเมินความถูกต้องของฟังก์ชันหลัก เช่น การลงทะเบียนผู้ใช้งาน การเข้าสู่ระบบ และการแสดงผลข้อมูลจากอุปกรณ์ IoT บนแดชบอร์ด นอกจากนี้ ได้ใช้เครื่องมือ Cypress

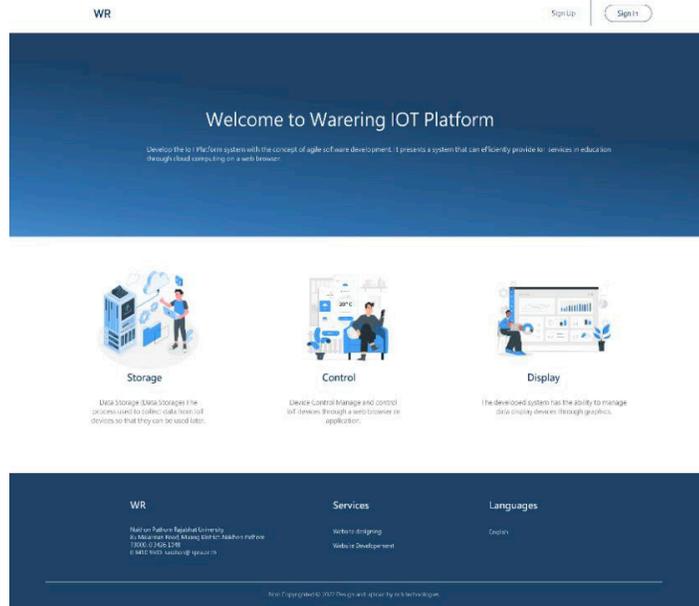
ซึ่งเป็นระบบทดสอบแบบ End-to-End ที่ใช้สำหรับเว็บแอปพลิเคชัน เพื่อดำเนินการทดสอบอัตโนมัติในมุมมองของผู้ใช้งานจริง ในส่วนของการทดสอบเชิงประสิทธิภาพและเครือข่าย ได้ใช้ Apache JMeter สำหรับการจำลองภาระงาน (Load Testing) เพื่อตรวจสอบความสามารถในการรองรับผู้ใช้งานจำนวนมากพร้อมกัน รวมถึงการทดสอบความทนทานของระบบภายใต้ภาระสูง ขณะเดียวกันได้ใช้ Wireshark ซึ่งเป็นเครื่องมือวิเคราะห์แพ็กเก็ตข้อมูลแบบโอเพ่นซอร์ส เพื่อตรวจสอบและวิเคราะห์การสื่อสารระหว่างอุปกรณ์ IoT กับเครือข่ายอย่างละเอียด ทั้งนี้ข้อมูลที่ได้จากกระบวนการทดสอบทั้งหมดจะถูกจัดเก็บและวิเคราะห์เพื่อระบุข้อบกพร่องหรือจุดที่ควรปรับปรุง ก่อนนำไปสู่การแก้ไขและปรับแต่งระบบให้พร้อมสำหรับการใช้งานจริงในสภาพแวดล้อมทางการศึกษา

5. การประเมินคุณภาพและความน่าใช้ของแพลตฟอร์ม ผู้วิจัยได้สำรวจความคิดเห็นจากกลุ่มผู้ใช้เป้าหมายจำนวน 30 คน ประกอบด้วยอาจารย์และนักศึกษาจากสาขาวิชาเทคโนโลยีสารสนเทศ (21 คน) และเทคโนโลยีคอมพิวเตอร์ (9 คน) โดยผู้เข้าร่วมได้ทดลองใช้งานแพลตฟอร์มจริงในบริบทของการเรียนการสอน การเก็บข้อมูลดำเนินการผ่านแบบสอบถามออนไลน์ (Google Forms) ประกอบด้วยคำถามแบบมาตราส่วนประเมินค่า 5 ระดับ (1 = พึงพอใจน้อยที่สุด ถึง 5 = พึงพอใจมากที่สุด) และยังมีคำถาม

ปลายเปิดไว้สำหรับข้อเสนอแนะเพิ่มเติม แบบสอบถามได้รับการตรวจสอบความตรงเชิงเนื้อหาโดยผู้ทรงคุณวุฒิ 3 ท่าน ซึ่งมีความเชี่ยวชาญด้านเทคโนโลยีการศึกษา เทคโนโลยีสารสนเทศ และวิศวกรรมซอฟต์แวร์ เพื่อให้มั่นใจว่าเครื่องมือมีความเหมาะสมและตรงตามวัตถุประสงค์ หัวข้อในการประเมินครอบคลุม 4 ด้านหลัก ได้แก่ (1) ความสะดวกในการใช้งาน (2) ความเหมาะสมของฟังก์ชัน (3) ประสิทธิภาพของระบบ และ (4) ความพึงพอใจโดยรวม ทั้งหมด 11 ข้อ โดยผลการประเมินจะถูกวิเคราะห์ด้วยสถิติเชิงพรรณนาและนำไปใช้เป็นข้อมูลในการปรับปรุงระบบในอนาคต

ผลการวิจัย

1. ผลการศึกษาข้อจำกัดของแพลตฟอร์ม IoT แบบเรียลไทม์ในปัจจุบัน และช่องว่างขององค์ความรู้ในบริบทการเรียนรู้ระดับอุดมศึกษาพบว่าแพลตฟอร์ม IoT ที่นิยมในปัจจุบัน เช่น Node-RED, ThingsBoard และ Blynk มีข้อจำกัดในการใช้งานภายในบริบทการเรียนรู้ในด้านความยุ่งยากในการใช้งานความซับซ้อนในการตั้งค่าและการจัดการสิทธิ์ผู้ใช้ ข้อจำกัดด้านการปรับแต่งและการรองรับผู้ใช้งานจำนวนมากส่งผลให้แพลตฟอร์มเหล่านี้ไม่ตอบสนองความต้องการในสภาพแวดล้อมการเรียนการสอน

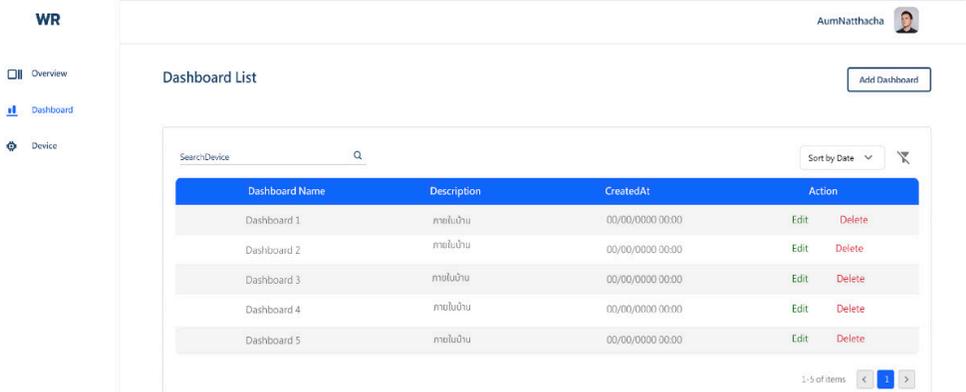


ภาพที่ 3 หน้าหลักของ WARERING Platform Homepage

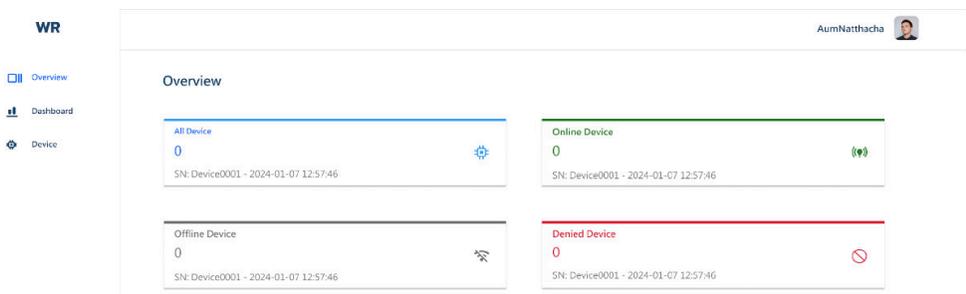
2. ผลการออกแบบและพัฒนาแพลตฟอร์ม WARERING สามารถตอบสนองความต้องการใช้งานในบริบทของการเรียนรู้ได้อย่างมีประสิทธิภาพ โดยการพัฒนาภายใต้แนวทางอจีลล์แบบสกรีม ซึ่งเน้นการพัฒนาแบบวนรอบและการปรับปรุงจากข้อเสนอแนะของผู้ใช้งานอย่างต่อเนื่อง ระบบรองรับการแสดงผลข้อมูลจากเซ็นเซอร์แบบเรียลไทม์ ผ่าน WebSocket และการควบคุมอุปกรณ์ผ่านโปรโตคอล MQTT พร้อมการจัดเก็บข้อมูลด้วย MongoDB และ Redis เพื่อเพิ่มความเร็วในการเข้าถึงและความเสถียรของระบบ ในส่วนของส่วนติดต่อผู้ใช้หน้าแรกของแพลตฟอร์ม (ภาพที่ 3) แสดงภาพรวมของระบบและการเข้าถึงฟังก์ชันหลัก ขณะที่หน้าจอบริหารจัดการอุปกรณ์ (ภาพที่ 4) แสดงเมนูสำหรับ เพิ่ม แก้ไข หรือลบอุปกรณ์ที่เชื่อมต่อส่วนหน้าจอแสดงอุปกรณ์ (ภาพที่ 5) แสดงสถานะของอุปกรณ์แต่ละตัวแบบเรียลไทม์ และหน้าจอแดชบอร์ด (ภาพที่ 6) รองรับการเพิ่มวิดเจ็ตรูปแบบต่าง ๆ เช่น ตัววัดอุณหภูมิกล้องข้อความ

ปุ่มควบคุม สวิตช์ และกราฟข้อมูล ซึ่งผู้ใช้สามารถปรับแต่งให้เหมาะสมกับรูปแบบการใช้งานได้

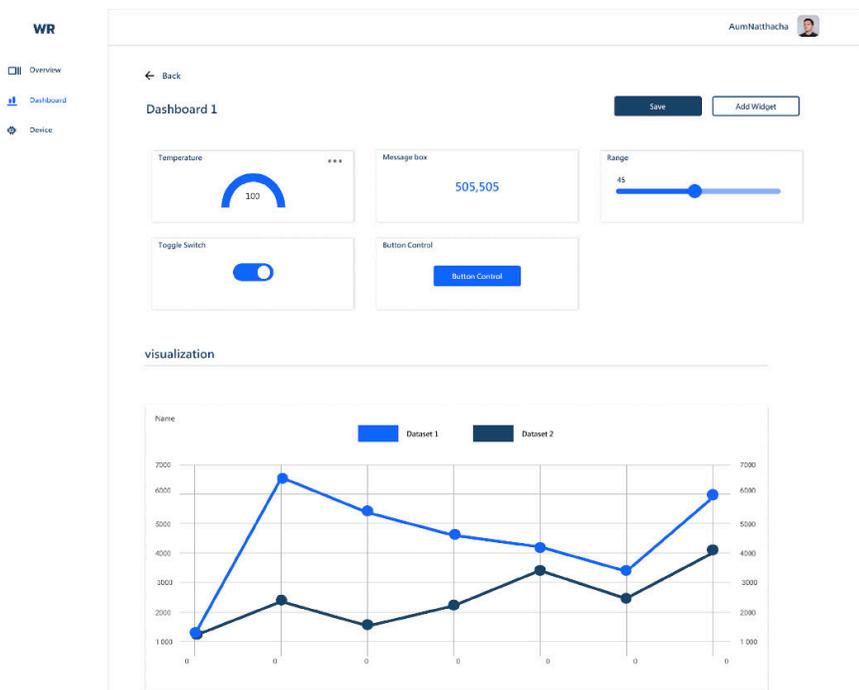
3. ผลการประเมินประสิทธิภาพ ฟังก์ชันการทำงาน และความสามารถในการเชื่อมต่อของแพลตฟอร์ม WARERING การประเมินประสิทธิภาพของแพลตฟอร์ม WARERING ดำเนินการโดยใช้เครื่องมือ Apache JMeter เพื่อจำลองผู้ใช้งานพร้อมกันในระดับต่าง ๆ ได้แก่ 50, 100, 500 และ 1,000 คน เป็นระยะเวลา 10 นาทีต่อรอบการทดสอบ ผลการทดสอบพบว่า จากตารางที่ 1 เมื่อมีผู้ใช้งานพร้อมกันไม่เกิน 500 คน ระบบสามารถรักษาเวลาในการตอบสนองเฉลี่ยให้อยู่ที่ประมาณ 480 มิลลิวินาที และมีอัตราความผิดพลาดเพียง 0.2% ซึ่งอยู่ในเกณฑ์ที่ยอมรับได้ อย่างไรก็ตาม เมื่อจำนวนผู้ใช้งานเพิ่มขึ้นเป็น 1,000 คน พบว่าเวลาในการตอบสนองเฉลี่ยเพิ่มขึ้นเป็น 800 มิลลิวินาที และอัตราความผิดพลาดเพิ่มขึ้นเป็น 2.5% ซึ่งสะท้อนถึงขีดจำกัดของระบบในเวอร์ชันปัจจุบัน



ภาพที่ 4 หน้าจอระบบบริหารจัดการอุปกรณ์



ภาพที่ 5 หน้าจอแสดงอุปกรณ์



ภาพที่ 6 หน้าจอแดชบอร์ดการตรวจสอบอุปกรณ์

ตารางที่ 1 การประเมินประสิทธิภาพการทำงานของระบบ

จำนวนผู้ใช้งานพร้อมกัน	เวลาในการตอบสนองเฉลี่ย (มิลลิวินาที)	อัตราความผิดพลาด (%)	ผลการประเมิน
50	250	0%	ผ่าน
100	350	0%	ผ่าน
500	480	0.2%	ผ่าน (มีความล่าช้าเล็กน้อย)
1,000	800	2.5%	ควรปรับปรุง

ในด้านการทดสอบฟังก์ชันการทำงาน ได้ใช้เครื่องมือ Cypress สำหรับทดสอบแบบ End-to-End ของระบบพบว่า ฟังก์ชันหลัก เช่น การลงทะเบียน ผู้ใช้งาน การเข้าสู่ระบบ การควบคุมอุปกรณ์ และการแสดงผลสถานะอุปกรณ์ IoT ทำงานได้ถูกต้องเกือบทั้งหมด ยกเว้นฟังก์ชันการแสดงผลข้อมูลแบบเรียลไทม์ที่มีความล่าช้าเฉลี่ยมากกว่า 5 วินาที ในบางช่วงเวลา ผลการทดลองแสดงในตารางที่ 2 ซึ่งเป็นประเด็นที่อยู่ระหว่างการปรับปรุงในการพัฒนารุ่นถัดไป นอกจากนี้ได้ทำการประเมินประสิทธิภาพด้านการเชื่อมต่อและการสื่อสารข้อมูลของระบบ โดยใช้เครื่องมือ Wireshark แสดงผลในตารางที่ 3 เพื่อวิเคราะห์การรับส่งข้อมูลผ่านโปรโตคอล MQTT

และ WebSocket ผลการตรวจสอบพบว่า การเชื่อมต่ออุปกรณ์ทำได้สำเร็จ 100% โดยไม่มีการสูญเสียแพ็กเก็ต ความเร็วเฉลี่ยในการส่งข้อมูลผ่าน MQTT อยู่ที่ 120 มิลลิวินาที และค่าความหน่วงของ WebSocket ต่ำกว่า 100 มิลลิวินาที โดยมีผลอยู่ในระดับดีมาก ในส่วนของสมรรถนะการส่งข้อมูลในระบบเครือข่ายโดยรวม ผลการทดสอบในตารางที่ 4 พบว่า อัตราการสูญเสียแพ็กเก็ตอยู่ที่ 0.1% ค่าความหน่วงเฉลี่ยอยู่ที่ 120 มิลลิวินาที และอัตราการใช้แบนด์วิดท์ อยู่ในเกณฑ์ต่ำกว่ามาตรฐาน ทั้งหมดนี้แสดงให้เห็นว่าระบบมีความเสถียรและประสิทธิภาพเพียงพอต่อการใช้งานในบริบทของการเรียนรู้ด้าน IoT ในระดับ อุดมศึกษา

ตารางที่ 2 ผลการทดสอบฟังก์ชันการทำงานของระบบ

ระบบที่ทดสอบ (ฟังก์ชัน)	จำนวนกรณีทดสอบ	ผ่าน	ล้มเหลว	หมายเหตุ
การลงทะเบียนและเข้าสู่ระบบของผู้ใช้	10	10	0	
การแสดงผลแดชบอร์ดแบบเรียลไทม์	12	11	1	5+ second delay
การแสดงผลสถานะของอุปกรณ์ IoT	8	8	0	
การควบคุมอุปกรณ์ IoT ผ่านหน้าเว็บเพจ	10	10	0	

ตารางที่ 3 ผลการตรวจสอบความสามารถในการเชื่อมต่อ

กรณีทดสอบ	ผลการทดสอบ	หมายเหตุ
การเชื่อมต่ออุปกรณ์ผ่าน MQTT Broker	เชื่อมต่อสำเร็จ 100%	ไม่มีการสูญเสียแพ็กเก็ต
ความเร็วเฉลี่ยในการส่งข้อมูลผ่าน MQTT	120 มิลลิวินาที	อยู่ในเกณฑ์ที่ยอมรับได้
การสื่อสารแบบเรียลไทม์ผ่าน WebSocket	ค่าความหน่วงเฉลี่ย <100 มิลลิวินาที	แสดงข้อมูลได้รวดเร็วมาก

ตารางที่ 4 ผลการทดสอบสมรรถนะการส่งข้อมูลในระบบเครือข่าย

รายการทดสอบ	ผลการทดสอบจริง	เกณฑ์การประเมิน	ผลการประเมิน
อัตราการสูญเสียแพ็กเกต	0.1%	≤ 1%	ผ่าน
ค่าความหน่วงเฉลี่ย (Latency)	120 มิลลิวินาที	≤ 200 มิลลิวินาที	ผ่าน
การใช้แบนด์วิดท์	250 กิโลไบต์/วินาที	≤ 500 กิโลไบต์/วินาที	ผ่าน

4. ผลการวิเคราะห์แบบสอบถามความพึงพอใจ จากผู้ใช้งานจำนวน 30 คน ดังแสดงในตารางที่ 5 พบว่า ผู้ใช้งานมีความพึงพอใจต่อระบบในภาพรวมอยู่ในระดับมากที่สุด โดยมีคะแนนเฉลี่ยรวมเท่ากับ 4.45 และค่าเบี่ยงเบนมาตรฐานเท่ากับ 0.53 เมื่อพิจารณารายละเอียด พบว่า ด้านความรวดเร็วในการเรียนรู้การใช้งานระบบ ได้รับคะแนนเฉลี่ยสูงสุดคือ 4.63 (ค่าเบี่ยงเบนมาตรฐาน = 0.34) แสดงให้เห็นว่าผู้ใช้งานสามารถทำความเข้าใจระบบได้อย่างรวดเร็ว ขณะที่ประเด็นอื่น ๆ เช่น ความสะดวกในการเข้าถึง

ข้อมูล ความเสถียรของระบบ ความน่าเชื่อถือของข้อมูลและการออกแบบส่วนติดต่อผู้ใช้ได้รับการประเมินในระดับ “มากที่สุด” เช่นกัน ทั้งนี้ แบบสอบถามความพึงพอใจได้รับการออกแบบ โดยอ้างอิงจากหลักเกณฑ์การประเมินระบบสารสนเทศในด้านต่าง ๆ ได้แก่ ความง่ายในการใช้งาน ความถูกต้อง ความเร็วในการประมวลผล ความเสถียรของระบบ และความเหมาะสมต่อการใช้งานในบริบททางการศึกษา โดยได้ผ่านการตรวจสอบความตรงเชิงเนื้อหาจากผู้เชี่ยวชาญ ก่อนนำไปใช้เก็บข้อมูลจริง

ตารางที่ 5 ผลการประเมินความพึงพอใจ

ประเด็นที่ประเมินความพึงพอใจ	คะแนนเฉลี่ย	ค่าเบี่ยงเบนมาตรฐาน	การแปลผล
1. ความรวดเร็วในการเรียนรู้การใช้งานระบบ	4.63	0.34	มากที่สุด
2. ความสะดวกในการเข้าถึงข้อมูล	4.47	0.50	มากที่สุด
3. ความถูกต้องของการแสดงผลข้อมูล	4.43	0.56	มากที่สุด
4. ความเสถียรของระบบระหว่างการใช้งาน	4.40	0.55	มากที่สุด
5. ความน่าเชื่อถือของข้อมูลที่แสดงผล	4.50	0.52	มากที่สุด
6. ความชัดเจนของเมนูและคำอธิบายภายในระบบ	4.38	0.60	มากที่สุด
7. ความเหมาะสมของฟังก์ชันที่มีในระบบ	4.42	0.51	มากที่สุด
8. ความรวดเร็วในการประมวลผลและแสดงผลข้อมูล	4.44	0.49	มากที่สุด
9. ความสามารถในการโต้ตอบกับข้อมูลแบบอินเทอร์แอคทีฟ	4.41	0.54	มากที่สุด
10. ความพึงพอใจต่อการออกแบบส่วนติดต่อผู้ใช้	4.23	0.59	มากที่สุด
11. ความรวดเร็วในการเรียนรู้การใช้งานระบบ	4.63	0.34	มากที่สุด
คะแนนรวม	4.45	0.53	มากที่สุด

สรุปและอภิปรายผลการวิจัย

ผลการวิจัยสอดคล้องกับวัตถุประสงค์ทั้ง 4 ข้อ โดยข้อแรกซึ่งศึกษาข้อจำกัดของแพลตฟอร์ม IoT ที่ใช้งานอยู่ในปัจจุบัน เช่น Node-RED, ThingsBoard และ Blynk พบว่ายังมีความซับซ้อนทางเทคนิคไม่เหมาะกับการใช้งานในชั้นเรียน โดยเฉพาะในด้านการตั้งค่าการจัดการสิทธิ์ และการปรับแต่งระบบ

สำหรับวัตถุประสงค์ข้อที่ 2 เกี่ยวกับการออกแบบและพัฒนาแพลตฟอร์ม WARERING ผลการพัฒนาแสดงให้เห็นถึงการวางแผนอย่างเป็นระบบ โดยเลือกใช้เทคโนโลยีที่เหมาะสมกับการเรียนรู้เชิงปฏิบัติ เช่น MQTT, WebSocket, Redis และ Kubernetes ทำให้ระบบสามารถสื่อสารแบบเวลาจริง รองรับผู้ใช้งานจำนวนมาก และติดตั้งได้ง่าย แม้ผู้ใช้ไม่มีความรู้ด้านเทคนิค

ในวัตถุประสงค์ข้อที่สาม การประเมินประสิทธิภาพของระบบด้วยเครื่องมือมาตรฐาน พบว่า WARERING รองรับผู้ใช้งานพร้อมกันได้สูงสุด 500 ราย มีค่าเวลาตอบสนองเฉลี่ยไม่เกิน 500 มิลลิวินาที และอัตราความผิดพลาดต่ำกว่า 1% แสดงถึงความเสถียรในการใช้งานจริง ส่วนในข้อสุดท้ายเกี่ยวกับความพึงพอใจของผู้ใช้ พบว่า ผู้ใช้งานให้คะแนนอยู่ในระดับมาก โดยเฉพาะด้านความสะดวกในการใช้งาน และความเข้าใจง่ายของระบบนอกจากนี้ การประยุกต์ใช้แนวทางการพัฒนาแบบบอจี้ร่วมกับกรอบการทำงาน

เอกสารอ้างอิง

- ธงชัย เจือจันทร์ ทวีวัฒน์ มุลจัด และ เอกธณช เหลืองศิริวรรณ (2566). TMAC: โพรโทคอลชั้นแมคเพื่อการสื่อสารระยะไกล สำหรับโพรโทคอลเอ็มคิวทีทีบน เครือข่ายลอรา. *วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยมหาสารคาม*, **42 (4)**, 108-117.
- Bhaskaran, H. S., Gordon, M., and Neethirajan, S. (2024). Development of a cloud-based IoT system for livestock health monitoring using AWS and python. *Smart Agricultural Technology*, **9**, 100524.
- Bhimanapati, V., Goel, P., and Jain, U. (2024). Leveraging Selenium and Cypress for Comprehensive Web Application Testing. *Journal of Quantum Science and Technology*, **1**, 66-79.
- Boca, L. L., Ciortea, E. M., Boghean, C., Begov-Ungur, A., Boghean, F., and Dădărlat, V. T. (2023). An IoT System Proposed for Higher Education: Approaches and Challenges in Economics, Computational Linguistics, and Engineering. *Sensors*, **23 (14)**, 13-21.

แบบสกริม โดยเฉพาะในขั้นตอน Sprint Review และ Retrospective ส่งผลให้ทีมพัฒนาสามารถปรับปรุงระบบได้ตรงจุดและทันต่อความต้องการของผู้ใช้ สอดคล้องกับงานวิจัยที่ชี้ว่าแนวทางบอจี้ช่วยลดเวลา ในการพัฒนา และเพิ่มความพึงพอใจของผู้ใช้งาน ทั้งนี้ ยังมีข้อจำกัดด้านโครงสร้างพื้นฐานที่ควรปรับปรุงเพื่อรองรับการขยายระบบในอนาคต

ข้อเสนอแนะ

จากผลการทดลองและการประเมินแพลตฟอร์ม WEARING ที่พัฒนาขึ้นผู้วิจัยมีข้อเสนอแนะเพื่อการพัฒนา และปรับปรุงในอนาคต ดังนี้

1. ปรับปรุงส่วนติดต่อผู้ใช้ ให้มีความทันสมัยและใช้งานง่ายยิ่งขึ้น เพื่อเสริมสร้างประสบการณ์ผู้ใช้ที่ดีขึ้น
2. เพิ่มสมรรถนะของระบบให้รองรับผู้ใช้งานจำนวนมากขึ้น
3. ขยายการทดสอบสู่สภาพแวดล้อมจริงที่มีผู้ใช้งานมากขึ้น และครอบคลุมกลุ่มสาขาวิชาอื่น เพื่อประเมินความยืดหยุ่นของระบบในบริบทที่หลากหลาย
4. ประยุกต์ใช้เทคโนโลยีวิเคราะห์ขั้นสูง เช่น Machine Learning และการประมวลผลข้อมูลแบบเรียลไทม์ เพื่อเพิ่มศักยภาพการใช้งานในระดับเชิงลึกและเชิงประยุกต์

- Böhm, S., and Wirtz, G. (2022). Cloud-Edge Orchestration for Smart Cities: A Review of Kubernetes based Orchestration Architectures. **EAI Endorsed Transactions on Smart Cities**, 6 (18), 1-19.
- Eyada, M., Saber, W., El Genidy, M., and Amer, F. (2020). Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments. **IEEE Access**, 8, 110656-110668.
- Ghashim, I. A., and Arshad, M. (2023). Internet of Things (IoT)-Based Teaching and Learning: Modern Trends and Open Challenges. **Sustainability**, 15 (21), 1-19.
- Han, X. (2021). A Study of Performance Testing in Configurable Software Systems. **Journal of Software Engineering and Applications**, 14 (9), 474-492.
- Hong, S. S., Lee, J., Chung, S., and Kim, B. (2023). Fast Real-Time Data Process Analysis Based on NoSQL for IoT Pavement Quality Management Platform. **Applied Sciences**, 13 (1), 3-18.
- Hwang, K., Jung, I. H., and Lee, A. M. (2022). Monitoring of MQTT-based Messaging Server. **Webology**, 19 (1), 4724-4735.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). **Manifesto for Agile Software Development**. Retrieved from October 12, 2024, <https://agilemanifesto.org/>.
- Mdemaya, G. B. J., Ndadjji, M. M. Z., Sindjoung, M. L. F., and Velempini, M. (2024). Efficient Load-Balancing and Container Deployment for Enhancing Latency in an Edge Computing-Based IoT Network Using Kubernetes for Orchestration. **International Journal of Advanced Computer Science and Applications**, 14 (9), 474-492.
- Pahl, C., El Ioini, N., Helmer, S., and Lee, B. (2018). **An Architecture Pattern for Trusted Orchestration in IoT Edge Clouds**. Retrieve from October 12, 2024, <https://doi.org/10.1109/FMEC.2018.8364046>.