

Received: 30 เม.ย. 2569

Revised: 2 มิ.ย. 2569

Accepted: 19 ส.ค. 2569

การพัฒนาเว็บแอปพลิเคชันระบบสารสนเทศโรงพยาบาล
สำหรับงานบันทึกข้อมูลการรักษาและการวินิจฉัยผู้ป่วยนอก

DEVELOPMENT OF A WEB-BASED HOSPITAL INFORMATION SYSTEM
FOR OUTPATIENT TREATMENT AND DIAGNOSIS RECORDING

เจษฎาภรณ์ สุธะวงศ์, ฐิติวัธส์ เพชรแทน, ธนวัฒน์ ปอกกันทา, สมนึก สินธุปวน,
กิตติกร หาญตระกูล และ พยุงศักดิ์ เกษมสำราญ*
สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้

Chetsadporn Suyawong, Thitiwat Petthaen, Tanawat Pokkantha,
Somnuek Sinthupoun, Kittikorn Hantrakul and Payungsak Kasemsumran*
Computer Science Department, Faculty of Science, Maejo University

*Corresponding author: payungsak.kae@gmail.com

Abstract

This research aims to develop a web-based Hospital Information System (HIS) for the outpatient department to replace the legacy desktop system, which has accessibility limitations from multiple service points. A key highlight of this study is the development of a system focused on centralized, real-time workflow integration utilizing a three-tier architecture. The system covers seven main modules: 1) Doctor Screen, 2) Nurse Counter, 3) Patient Screening (PTScreen), 4) Treatment, 5) Medical History, 6) Doctor Summary, and 7) Patient Profile.

The system was developed using React.js and Node.js in conjunction with a MySQL database. The system's performance evaluation was conducted through test cases. The results revealed that the system achieved 100% functional accuracy across all test cases. This demonstrates that the system can accurately retrieve and display medical history data from the existing HIS via an API in real time. Consequently, the system helps reduce data entry redundancy for healthcare personnel and can be practically implemented to tangibly elevate service efficiency in the outpatient department.

Keywords: Hospital Information System; Outpatient Department; Web Application; Three-tier Architecture; Functional Accuracy; API Data Integration

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบสารสนเทศในโรงพยาบาล (Hospital Information System: HIS) สำหรับงานผู้ป่วยนอก ในรูปแบบเว็บแอปพลิเคชัน เพื่อทดแทนระบบเดสก์ท็อปเดิมที่มีข้อจำกัดด้านการเข้าถึงจากหลายจุดบริการ จุดเด่นของงานวิจัยนี้คือการพัฒนาเว็บที่มุ่งเน้นการเชื่อมโยงกระบวนการทำงานแบบรวมศูนย์แบบเรียลไทม์ผ่านโครงสร้างสถาปัตยกรรมแบบทรีเทียร์ ครอบคลุม 7 ส่วนงานหลัก ได้แก่ 1) ส่วนงานบันทึกข้อมูลของแพทย์ 2) ส่วนงานบันทึกข้อมูลของพยาบาล 3) ส่วนงานซักประวัติ 4) ส่วนงานการรักษาและหัตถการ 5) ส่วนงานประวัติการรักษา 6) ส่วนงานสรุปผลการรักษา และ 7) ส่วนงานข้อมูลผู้ป่วย ระบบพัฒนาโดยใช้ React.js และ Node.js ร่วมกับฐานข้อมูล MySQL การประเมินประสิทธิภาพของระบบดำเนินการผ่านการทดสอบตามกรณีใช้งานผลการทดสอบพบว่า ระบบมีความถูกต้องเชิงฟังก์ชันอยู่ที่ร้อยละ 100 จากกรณีทดสอบทั้งหมด แสดงให้เห็นว่าระบบสามารถดึงและแสดงผลข้อมูลประวัติการรักษาจากระบบ HIS เดิมผ่าน API ได้แบบเรียลไทม์อย่างถูกต้อง ช่วยลดความซ้ำซ้อนในการบันทึกข้อมูลของบุคลากร และสามารถนำไปประยุกต์ใช้เพื่อยกระดับประสิทธิภาพการให้บริการในหน่วยงานผู้ป่วยนอก

คำสำคัญ: ระบบสารสนเทศโรงพยาบาล; งานผู้ป่วยนอก; เว็บแอปพลิเคชัน; สถาปัตยกรรมแบบทรีเทียร์; ความถูกต้องเชิงฟังก์ชัน; การเชื่อมต่อข้อมูลผ่าน API

บทนำ

ในปัจจุบัน แผนกผู้ป่วยนอก (Outpatient Department: OPD) ของสถานพยาบาลต้องเผชิญกับปริมาณผู้รับบริการที่เพิ่มสูงขึ้นอย่างต่อเนื่อง ทำให้กระบวนการจัดการข้อมูลทางคลินิกตั้งแต่การลงทะเบียน ซักประวัติ ไปจนถึงการวินิจฉัยและวางแผนการรักษามีความซับซ้อนและท้าทายมากขึ้น การแก้ปัญหาดังกล่าวจึงจำเป็นต้องอาศัยระบบเทคโนโลยีสารสนเทศโรงพยาบาล (Hospital Information System: HIS) ที่มีมาตรฐานเข้ามาช่วยบูรณาการข้อมูลแบบรวมศูนย์ ซึ่งจะช่วยลดความซ้ำซ้อนในการบันทึกข้อมูลของเจ้าหน้าที่ในแต่ละจุดบริการ ลดความคลาดเคลื่อนทางเอกสาร และทำให้บุคลากรทางการแพทย์สามารถสืบค้นประวัติผู้ป่วยได้อย่างรวดเร็ว ถูกต้อง และทันทั่วถึง การพัฒนาระบบสารสนเทศที่ได้มาตรฐานนี้ ไม่เพียงแต่ช่วยลดระยะเวลาการรอคอยและเพิ่มประสิทธิภาพในการให้บริการ แต่ยังเป็นโครงสร้างพื้นฐานสำคัญที่ช่วยสนับสนุนการตัดสินใจทางคลินิกเพื่อความปลอดภัยสูงสุดของผู้ป่วย (สมาคมเวชสารสนเทศไทย, 2564)

ระบบสารสนเทศโรงพยาบาลในรูปแบบเดสก์ท็อปแอปพลิเคชัน (Desktop Application) จะยังมีการใช้งานอยู่ในหลายหน่วยงาน แต่ปัจจุบันพบข้อจำกัดทางเทคนิคที่เป็นอุปสรรคต่อการทำงาน ความยุ่งยากในการติดตั้งและบำรุงรักษาซอฟต์แวร์แบบรายเครื่อง การปรับปรุงสถาปัตยกรรม

ระบบที่ทำได้ยากในระยะยาว และการจำกัดการเข้าถึงข้อมูลทางคลินิกได้เฉพาะบนเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมไว้เท่านั้น ข้อจำกัดด้านโครงสร้างพื้นฐานแบบแยกส่วนและไม่ยืดหยุ่นนี้ทำให้บุคลากรทางการแพทย์ทำงานได้ไม่ต่อเนื่อง และเป็นอุปสรรคสำคัญต่อส่งต่อข้อมูลเพื่อพัฒนาคุณภาพการให้บริการสาธารณสุข (กองยุทธศาสตร์และแผนงาน สำนักงานปลัดกระทรวงสาธารณสุข, 2565)

ด้วยเหตุนี้ ผู้วิจัยจึงได้พัฒนาระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอก ส่วนงานการบันทึกข้อมูลการรักษาและการวินิจฉัย ในรูปแบบเว็บแอปพลิเคชัน (Web Application) เพื่อปิดช่องว่างดังกล่าวและแก้ไขปัญหาด้านการเข้าถึงข้อมูลจากหลายจุดบริการผ่านเว็บเบราว์เซอร์ โดยไม่จำเป็นต้องติดตั้งโปรแกรมเพิ่มเติม ระบบที่พัฒนาขึ้นนี้มีความแตกต่างและโดดเด่นจากระบบทั่วไปโดยมุ่งเน้นการสร้างสถาปัตยกรรมแบบรวมศูนย์ที่ครอบคลุมกระบวนการทำงานตั้งแต่การซักประวัติผู้ป่วย การบันทึกข้อมูลของพยาบาลหน้าเคาน์เตอร์และแพทย์ การวินิจฉัยโรค ไปจนถึงการจัดทำเอกสารสรุปผลการรักษา พร้อมความสามารถในการดึงข้อมูลประวัติการรักษาจากระบบเดิม (Legacy HIS) ผ่าน API ได้ทันที เพื่อให้การดำเนินงานในหน่วยบริการผู้ป่วยนอกมีความต่อเนื่องถูกต้อง พร้อมรองรับการขยายผลไปสู่ฟังก์ชันการทำงานที่ซับซ้อน นอกจากนี้ สถาปัตยกรรมแบบสามชั้น (Three-tier Architecture) ยังเป็นโครงสร้างที่ได้รับการยอมรับในการพัฒนาระบบสารสนเทศทางการแพทย์ เนื่องจากมีการแยกส่วนแสดงผล (Presentation Tier) ส่วนประมวลผล (Logic Tier) และส่วนจัดการข้อมูล (Data Tier) (ITU Online, 2024) ออกจากกันอย่างชัดเจน ซึ่งช่วยยกระดับความปลอดภัยของข้อมูลและรองรับการขยายตัวของระบบได้อย่างมีประสิทธิภาพ (IBM, 2024)

วัตถุประสงค์ของการวิจัย

1. เพื่อพัฒนาระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอก (HIS-OPD) ในรูปแบบเว็บแอปพลิเคชัน ที่ครอบคลุมส่วนงานการบันทึกข้อมูลการรักษาและการวินิจฉัยโรค
2. เพื่อประเมินคุณภาพการทำงานของระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอกที่พัฒนาขึ้น ด้วยการทดสอบความถูกต้องเชิงฟังก์ชัน (Functional Testing)

การทบทวนวรรณกรรม

จากการทบทวนวรรณกรรมที่เกี่ยวข้องกับการพัฒนาระบบสารสนเทศทางคลินิก (Clinical Information System: CIS) และระบบดูแลสุขภาพบนเครือข่ายเว็บ (Web-based Healthcare System) พบว่า การเปลี่ยนผ่านไปสู่สถาปัตยกรรมเว็บและการเชื่อมโยงข้อมูลผ่านระบบอินเทอร์เฟซโปรแกรมประยุกต์ (Application Programming Interface: API) ช่วยเพิ่มความยืดหยุ่นและลดความคลาดเคลื่อนในกระบวนการดูแลผู้ป่วยได้อย่างมีนัยสำคัญ (Kuo, Kushniruk, & Borycki,

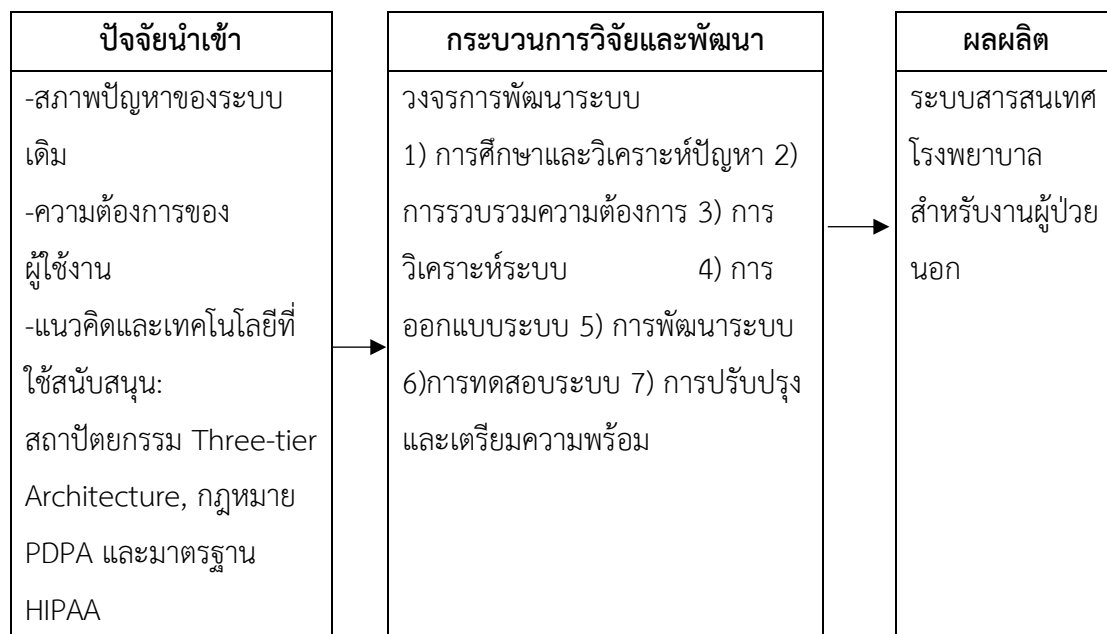
2020) ดังที่ นิษฐา หอมวุฒิวังศ์ และคณะ (2561) ได้นำเสนอว่าระบบสารสนเทศทางคลินิกที่ออกแบบให้รองรับการบันทึกข้อมูลการรักษาอย่างเป็นระบบ สามารถช่วยลดความผิดพลาดและเพิ่มประสิทธิภาพการทำงานของบุคลากรได้ นอกจากนี้ Hoyt และ Hersh (2018) รวมถึง Nelson และ Staggers (2018) ได้เน้นย้ำว่า ความท้าทายสำคัญของระบบสารสนเทศทางการแพทย์ในปัจจุบันคือการขาดการรวมศูนย์ข้อมูลและการทำงานที่แยกส่วนกันระหว่างจุดบริการ ส่งผลให้เกิดช่องว่างของงานวิจัยเนื่องจากระบบส่วนใหญ่ยังไม่สามารถเชื่อมโยงข้อมูลการคัดกรอง การซักประวัติ และการตรวจรักษาของแพทย์เข้าด้วยกันเป็นระบบนิเวศข้อมูลเดียวกันในรูปแบบเรียลไทม์

วรรณกรรมที่เกี่ยวข้องกับสถาปัตยกรรมแบบสามชั้น (Three-tier Architecture)
พบว่า เป็นแนวความคิดการออกแบบโครงสร้างซอฟต์แวร์ที่ได้รับการยอมรับในระดับสากล โดยมีหลักการสำคัญคือการแบ่งระบบออกเป็น 3 ชั้นหลักอย่างชัดเจน เพื่อแยกหน้าที่ความรับผิดชอบออกจากกันอย่างเด็ดขาด (IBM, 2024) เริ่มจากชั้นแรกคือ ส่วนที่ติดต่อกับผู้ใช้งาน (Presentation Tier) ซึ่งในปัจจุบันนิยมพัฒนาให้อยู่ในรูปแบบของเว็บแอปพลิเคชัน (Web application) เพื่อให้ผู้ใช้สามารถโต้ตอบกับระบบได้สะดวก ถัดมาคือ ส่วนประมวลผล (Logic Tier) ของระบบที่คอยรับคำสั่งมาประมวลผลตามตรรกะและเงื่อนไขต่าง ๆ และชั้นสุดท้ายคือ หรือชั้นจัดการฐานข้อมูล (Data Tier) ที่มีหน้าที่จัดเก็บข้อมูลทั้งหมดของระบบ (ITU Online, 2024) การแยกส่วนประกอบเหล่านี้ออกจากกันมีส่วนสำคัญด้านความปลอดภัยอีกด้วย เนื่องจากผู้ใช้งานหรืออุปกรณ์ปลายทางจะไม่สามารถเข้าถึงชั้นฐานข้อมูลได้โดยตรง ทุกคำสั่งจะต้องผ่าน ส่วนประมวลผล (Application Tier) เสมอ (Federal Office for Information Security (BSI), 2025) ซึ่งชั้นกลางนี้เองที่เป็นจุดควบคุมความถูกต้องเชิงฟังก์ชัน (Functional correctness) ทำให้ผู้พัฒนาระบบสามารถกำหนดกฎเกณฑ์ทางธุรกิจ เพื่อตรวจสอบว่าข้อมูลทุกอย่างถูกต้องตามข้อกำหนดก่อนที่จะอนุญาตให้บันทึกลงฐานข้อมูลจริง (ITU Online, 2024) เพื่อให้เห็นภาพการนำไปใช้งานจริงที่ชัดเจนยิ่งขึ้น หากเรานำสถาปัตยกรรมแบบทรีเทียร์มาประยุกต์ใช้กับระบบที่มีความซับซ้อนและต้องการความปลอดภัยสูงอย่างระบบสารสนเทศโรงพยาบาล โดยเฉพาะในส่วนของการผู้ป่วยนอก จะพบว่าการแบ่งชั้นระบบเช่นนี้ช่วยลดความผิดพลาดทางการแพทย์ได้อย่างมาก ซึ่งกระบวนการที่เป็นระบบระเบียบเช่นนี้สอดคล้องกับมาตรฐานการจัดการระบบสารสนเทศทางการแพทย์ที่เน้นความมั่นคงปลอดภัยของข้อมูลผู้ป่วยเป็นสำคัญ

วิธีการดำเนินงานวิจัย

งานวิจัยนี้ดำเนินการในรูปแบบการวิจัยและพัฒนา (Research and Development) โดยผู้วิจัยได้ประยุกต์ใช้วงจรการพัฒนากระบวนการ (Systems Development Life Cycle: SDLC) มาเป็น

กรอบแนวทางหลักในการดำเนินงาน เพื่อแก้ปัญหาคำซ้อนซ้อนของการอธิบายเนื้อหา และช่วยให้กระบวนการศึกษามีความเชื่อมโยงเป็นระบบตามหลักวิชาการ



รูปที่ 1 กรอบแนวคิดการวิจัย

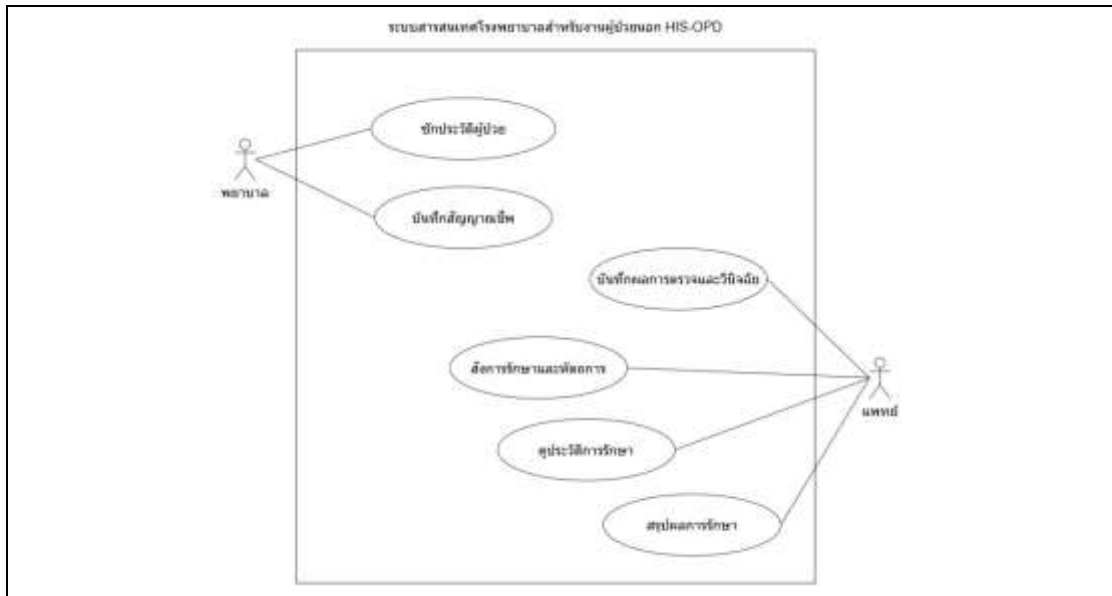
โดยแบ่งขั้นตอนการดำเนินงานออกเป็น 7 ขั้นตอน ดังนี้

1. ขั้นตอนการศึกษาและวิเคราะห์ปัญหาของระบบเดิม (System Study & Analysis) ผู้วิจัยทำการศึกษขั้นตอนการปฏิบัติงานจริงทางคลินิก (Clinical Workflow) ในหน่วยบริการผู้ป่วยนอก โดยใช้วิธีการสังเกตการณ์อย่างมีส่วนร่วม (Participant Observation) ในพื้นที่ปฏิบัติงานจริง และการสัมภาษณ์เชิงลึก (In-depth Interview) ร่วมกับบุคลากรทางการแพทย์ จากการศึกษาพบว่าระบบสารสนเทศโรงพยาบาลเดิมซึ่งมีโครงสร้างแบบเดสก์ท็อปแอปพลิเคชัน มีข้อจำกัดสำคัญคือข้อมูลถูกจัดเก็บแยกส่วน ไม่รองรับการเข้าถึงข้อมูลพร้อมกันจากหลายจุดไม่สะดวกต่อการบำรุงรักษาซอฟต์แวร์ที่เครื่องลูกข่าย (Client Machines) ปัญหาเหล่านี้ส่งผลให้การส่งต่อข้อมูลผู้ป่วยระหว่างหน้าเคาน์เตอร์พยาบาลและห้องตรวจแพทย์เกิดความล่าช้า

2. ขั้นตอนการรวบรวมความต้องการจากผู้ใช้งาน (Requirements Gathering) ใช้วิธีการจัดประชุมกลุ่มย่อย (Focus Group) ร่วมกับกลุ่มผู้ใช้งานหลักเพื่อรวบรวมความต้องการเชิงหน้าที่ (Functional Requirements) ประกอบด้วย แพทย์จำนวน 5 ท่าน และพยาบาลวิชาชีพจำนวน 10 ท่าน พบว่า ผู้ใช้งานต้องการระบบในรูปแบบเว็บแอปพลิเคชันที่สามารถรวบรวมข้อมูลคัดกรองสัญญาณชีพ ประวัติการรักษา และการสั่งหัตถการไว้ในแพลตฟอร์มเดียวกันอย่างรวมศูนย์ รวมถึงต้องสามารถเชื่อมโยงข้อมูลกับระบบฐานข้อมูลเดิม (Legacy HIS) แบบเรียลไทม์ได้

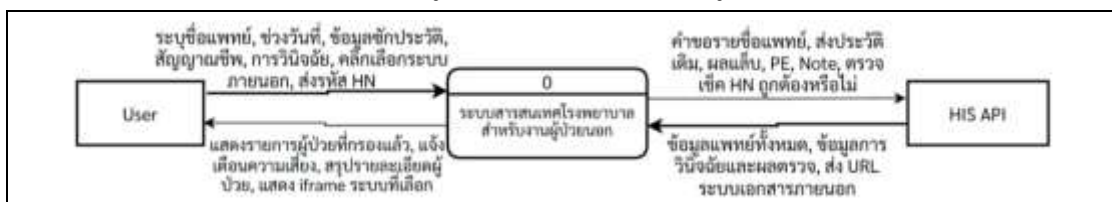
3. ขั้นตอนการวิเคราะห์ระบบ (System Analysis) การวิเคราะห์ระบบเป็นขั้นตอนสำคัญเนื่องจากก่อนพัฒนาระบบผู้วิจัยได้ศึกษาขั้นตอนการทำงานจริงในคลินิกเพื่อทำความเข้าใจว่าแต่ละ

คนมีบทบาทหน้าที่อย่างไร พบว่าผู้ใช้งานหลักมี 2 กลุ่ม คือ แพทย์ ที่ทำหน้าที่วินิจฉัยและสั่ง การ รักษา และ พยาบาล ที่ทำหน้าที่คัดกรองผู้ป่วยและประสานงานก่อนเข้าพบแพทย์ โดยทั้งสองกลุ่ม ต้องการระบบที่ใช้งานง่าย ลดขั้นตอนที่ซ้ำซ้อน และแสดงข้อมูลที่จำเป็นได้ครบในที่เดียว ดังรูปที่ 2



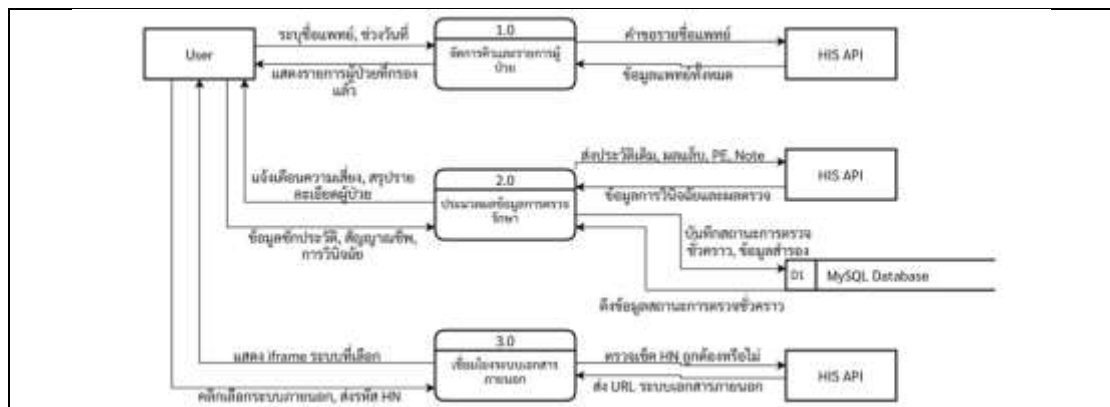
รูปที่ 2 แผนภาพกรณีการใช้งาน (Use Case Diagram) ของระบบสำหรับแพทย์และพยาบาล

4. ขั้นตอนการออกแบบระบบ (System Design) ระบบได้รับการออกแบบโดยใช้แนวคิด Three-tier Architecture ซึ่งแบ่งโครงสร้างออกเป็นสามชั้นการทำงานอย่างชัดเจน ได้แก่ ชั้นแสดงผลที่พัฒนาด้วย React.js ชั้นประมวลผลที่พัฒนาด้วย Node.js และชั้นจัดเก็บข้อมูลที่ใช้ MySQL การแบ่งโครงสร้างในลักษณะนี้ทำให้แต่ละชั้นทำงานได้อย่างอิสระต่อกัน ส่งผลให้ระบบมีความยืดหยุ่นสูง สามารถปรับเปลี่ยนหรือพัฒนาแต่ละส่วนได้โดยไม่กระทบต่อส่วนอื่น นอกจากนี้สถาปัตยกรรมดังกล่าวยังรองรับการเชื่อมต่อกับระบบ HIS เดิมผ่าน API ได้อย่างราบรื่น พร้อมทั้งช่วยลดความซับซ้อนในการบำรุงรักษาและดูแลระบบในระยะยาว ดังรูปที่ 3



รูปที่ 3 แผนภาพกระแสข้อมูลระดับ 0 (Context Diagram) ของระบบสารสนเทศโรงพยาบาลสำหรับงานผู้ป่วยนอก

นอกจากนี้ ยังให้ความสำคัญกับการไหลของข้อมูล (Data Flow) เพื่อให้โมดูลต่าง ๆ ทำงานเชื่อมโยงกันได้อย่างต่อเนื่อง ตั้งแต่การที่พยาบาลบันทึกสัญญาณชีพและประวัติผู้ป่วย ไปจนถึงการที่แพทย์วินิจฉัยและสั่งการรักษา ช่วยลดความซ้ำซ้อนในการทำงาน และสนับสนุนการตัดสินใจทางคลินิกของแพทย์และพยาบาลได้อย่างราบรื่น ดังรูปที่ 4



รูปที่ 4 แผนภาพกระแสข้อมูลระดับ 1 (DFD Level 1) ของระบบ

แสดงการไหลของข้อมูลระหว่างผู้ใช้งานและกระบวนการทำงานหลัก 3 ส่วน ได้แก่ การดูรายการผู้ป่วย การดูรายละเอียดผู้ป่วย และการแสดงระบบภายนอก โดยแต่ละส่วนเชื่อมต่อกับ HIS API ของระบบเพื่อดึงข้อมูลที่เกี่ยวข้องมาแสดงผลและมีการทำงานร่วมกับ D1: MySQL Database เพื่อทำหน้าที่สำรองข้อมูลการตรวจชั่วคราวและเก็บสถานะการใช้งานระบบ ลดปัญหาข้อมูลสูญหายระหว่างการเชื่อมต่อ API

5. ขั้นตอนการพัฒนากระบวน (System Development) ผู้วิจัยพัฒนาระบบตามรูปแบบสถาปัตยกรรม Three-tier Architecture ซึ่งแบ่งแยกส่วนการทำงานออกเป็น 3 ชั้น ได้แก่

5.1 ชั้นแสดงผล (Presentation Tier) พัฒนาด้วย React.js ร่วมกับ Tailwind CSS

5.2 ชั้นประมวลผล (Logic Tier) พัฒนาด้วย Node.js

5.3 ชั้นจัดเก็บข้อมูล (Data Tier) โดยมีการพัฒนา API เพื่อเชื่อมต่อระบบใหม่เข้ากับระบบ HIS เดิมของโรงพยาบาลในรูปแบบเรียลไทม์

6. ขั้นตอนการทดสอบระบบ ผู้วิจัยได้ดำเนินการประเมินคุณภาพระบบด้วยการทดสอบความถูกต้องเชิงฟังก์ชันโดยใช้สถานการณ์จำลองทางคลินิก ร่วมกับชุดข้อมูลทดสอบสมมติ จำนวน 110 กรณีทดสอบ ครอบคลุมทั้ง 7 ส่วนงาน โดยมีผู้วิจัยและตัวแทนบุคลากรทางการแพทย์ ผู้ใช้งานจริง ร่วมเป็นผู้ทำการทดสอบจำนวน 3 รอบ ซึ่งมีเกณฑ์การประเมินคือ ระบบจะต้องประมวลผลข้อมูลและแสดงการแจ้งเตือนเงื่อนไขความเสี่ยงทางคลินิก เช่น การแพ้ยาและค่าสัญญาณชีพผิดปกติ ได้ถูกต้องครบถ้วนตรงตามที่ออกแบบไว้ ผลจากการทดสอบพบข้อผิดพลาดเล็กน้อยในส่วนของการแสดงผลแบบอักษรในส่วนงาน Doctor Summary ซึ่งผู้วิจัยได้ดำเนินการแก้ไขในรหัสต้นฉบับ ทันทีจนระบบสามารถทำงานได้ถูกต้องร้อยละ 100

7. ขั้นตอนการปรับปรุงแก้ไขและเตรียมความพร้อมสำหรับการใช้งานจริงผู้วิจัยนำข้อบกพร่องจากระบบการทดสอบมาปรับปรุงประสิทธิภาพของซอฟต์แวร์ นอกจากนี้ ผู้วิจัยได้เพิ่มมาตรการด้านความมั่นคงปลอดภัยของข้อมูล เพื่อให้สอดคล้องกับพระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล

บุคคล พ.ศ. 2562 (ราชกิจจานุเบกษา, 2562) และมาตรฐานสากลด้านข้อมูลสุขภาพ (Health Insurance Portability and Accountability Act, 1996) ด้วยการกำหนดสิทธิ์การเข้าถึงข้อมูลตามบทบาทหน้าที่ (Role-Based Access Control: RBAC) ทำให้แพทย์แต่ละท่านสามารถเข้าถึงได้เฉพาะแฟ้มประวัติของผู้ป่วยที่อยู่ในความดูแลของตนเองเท่านั้น พร้อมทั้งเพิ่มระบบการเข้ารหัสข้อมูลและการขอตั้งรหัสผ่านใหม่ ผ่านระบบอีเมลเพื่อความปลอดภัยขั้นสูงสุด

ขั้นตอนวิธีของระบบ เพื่อให้การประมวลผลข้อมูลทางคลินิกมีความแม่นยำและปลอดภัย ผู้วิจัยได้ออกแบบขั้นตอนวิธีสำหรับส่วนงานที่เป็นศูนย์กลางการทำงานและมีการประมวลผลข้อมูลที่ซับซ้อน ดังนี้

ขั้นตอนวิธีที่ 1 การโหลดและบันทึกข้อมูลการตรวจแพทย์

1.1 ข้อมูลนำเข้า รหัสผู้ป่วย (HN), รหัสการมาตรวจ (VN), วันที่ตรวจ (visitDate), รหัสแพทย์ (doctorId)

1.2 ผลลัพธ์ ข้อมูลการตรวจที่บันทึกสำเร็จในระบบ (savedMedicalRecord)

1.3 กระบวนการทำงาน ระบบจะดึงข้อมูลผู้ป่วยและประวัติการตรวจเดิมจากระบบผ่าน API พร้อมตรวจสอบเงื่อนไขความเสี่ยงเพื่อแสดงการแจ้งเตือนอัตโนมัติ จากนั้นรับข้อมูลผลการตรวจจากแพทย์แล้วส่งกลับไปบันทึกยังระบบฐานข้อมูลหลักพร้อมสำรองข้อมูลชั่วคราว

1. เริ่มต้น
2. โหลดข้อมูลผู้ป่วยจาก HN และ VN ผ่าน API และตรวจสอบความครบถ้วนของข้อมูล
3. ตรวจสอบและแสดงการแจ้งเตือนอัตโนมัติ
 - 3.1 ตรวจสอบประวัติแพ้ยา
 - 3.2 ตรวจสอบยาค้างรับจากเภสัช
 - 3.3 ตรวจสอบสถานะการ Admit
 - 3.4 ตรวจสอบสถานะผู้ป่วยเสียชีวิต
4. โหลดข้อมูลประวัติการตรวจเดิม (Medical History และ PE Data) ผ่าน API ที่มีการยืนยันตัวตน (Secure API)
5. แพทย์กรอกข้อมูลการตรวจ (CC, PI, PE, Doctor Note, F/U & Investigate)
6. กดบันทึก: ระบบส่งข้อมูลผ่าน API (/api/save-doctor-screen และ /api/update-pe-screen) ไปยัง HIS
7. อัปเดตข้อมูลสถานะการตรวจชั่วคราวลงในหน่วยความจำเซสชัน (Secure Session State) ที่มีการเข้ารหัส
8. สิ้นสุด

ขั้นตอนวิธีที่ 2 ระบบตรวจสอบและแจ้งเตือนข้อมูลสัญญาณชีพผิดปกติ

2.1 ข้อมูลนำเข้า ข้อมูลสัญญาณชีพ อุณหภูมิ, ชีพจร, อัตราการหายใจ, ความดันโลหิต ที่บันทึกผ่านส่วนงาน Nurse Counter

2.2 ผลลัพธ์ การแสดงผลสถานะความเสี่ยงทางคลินิก บนหน้าจอ

2.3 กระบวนการทำงาน เพื่อเชื่อมโยงการทำงานย่อยเข้ากับระบบหลักแบบเรียลไทม์ ระบบจะนำค่าสัญญาณชีพที่พยาบาลบันทึก มาเปรียบเทียบกับเกณฑ์ค่ามาตรฐานสากลทางการแพทย์ด้วยฟังก์ชันทางคณิตศาสตร์ทันที หากพบค่าที่เข้าข่ายวิกฤต ระบบจะเปลี่ยนสีการแสดงผลเป็นสีแดง พร้อมส่งสัญญาณเตือนไปที่หน้าจอของส่วนงาน Doctor Screen โดยอัตโนมัติ เพื่อให้แพทย์สามารถรับทราบสถานการณ์และเตรียมการส่งต่อผู้ป่วยเข้าพบได้อย่างเร่งด่วน

1. เริ่มต้น
2. รับค่าข้อมูลสัญญาณชีพ อุณหภูมิ, ชีพจร, อัตราการหายใจ, ความดันโลหิต จากพยาบาลผ่านหน้าจอ Nurse Counter
3. ระบบนำค่าสัญญาณชีพมาเปรียบเทียบกับเกณฑ์ค่ามาตรฐานสากลทางการแพทย์ด้วยฟังก์ชันทางคณิตศาสตร์
4. ตรวจสอบเงื่อนไขความเสี่ยง
 - 4.1 หากค่าอยู่ในเกณฑ์ปกติ ระบบจะแสดงผลด้วยข้อความสีปกติ
 - 4.2 หากพบค่าที่เข้าข่ายวิกฤต ระบบจะเปลี่ยนสีการแสดงผลเป็นสีแดง
5. ส่งข้อมูลแจ้งเตือนสถานะวิกฤต ไปแสดงผลที่หน้าจอของส่วนงาน Doctor Screen โดยอัตโนมัติ เพื่อให้แพทย์เตรียมการ
6. บันทึกข้อมูลและสถานะความเสี่ยงลงสู่ฐานข้อมูลหลัก
7. สิ้นสุด

ผลการวิจัย

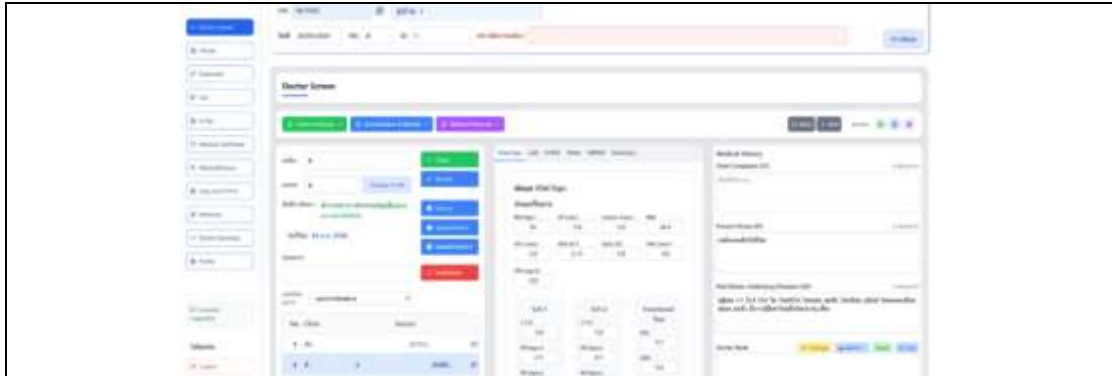
ผลการพัฒนาระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอก ส่วนงานการบันทึกข้อมูลการรักษาและการวินิจฉัย สามารถจำแนกผลการดำเนินงานออกเป็น 2 ส่วนหลัก เพื่อให้สอดคล้องกับวัตถุประสงค์ของการวิจัย ดังนี้

1. ผลการพัฒนาระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอก

ระบบที่พัฒนาขึ้นเป็นเว็บแอปพลิเคชันที่ทำงานร่วมกับฐานข้อมูลหลักของโรงพยาบาลผ่าน API สามารถใช้งานได้สมบูรณ์ครอบคลุม 7 ส่วนงานหลัก ดังนี้

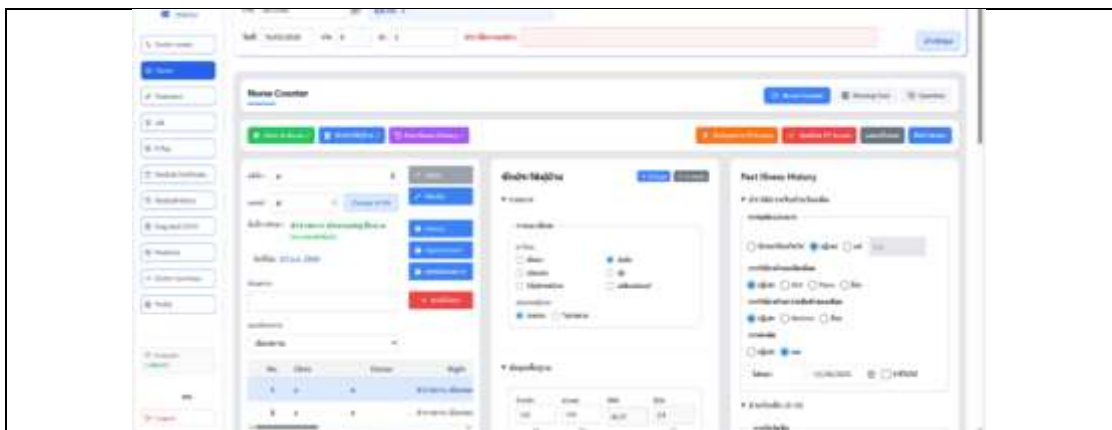
1.1 ส่วนงาน Doctor Screen คือหน้าจอสำหรับแพทย์ใช้งานในห้องตรวจ ตั้งแต่เปิดดูข้อมูลผู้ป่วยจนถึงสิ้นสุดการตรวจ ข้อมูลที่จำเป็นทั้งหมดจะแสดงขึ้นมาในที่เดียวทันที แพทย์

สามารถค้นหารหัสโรค สัณยา และบันทึกผลการตรวจ อาการ บันทึกการวินิจฉัย พร้อมระบบแจ้งเตือนอัตโนมัติหากพบความเสี่ยง เช่น ผู้ป่วยแพ้ยา หรือมีการแจ้งเตือนจากเภสัชกร ดังรูปที่ 5



รูปที่ 5 ตัวอย่างหน้าจอการบันทึกข้อมูลแพทย์

1.2 ส่วนงาน Nurse Counter คือหน้าจอสำหรับพยาบาลใช้งานก่อนผู้ป่วยเข้าพบแพทย์ เพื่อบันทึกสัญญาณชีพ (Vital Signs) เช่น ความดันโลหิต ชีพจร อุณหภูมิ และประเมินระดับความเสี่ยงของผู้ป่วยให้อัตโนมัติ เมื่อเลือกห้องตรวจและกดบันทึก ข้อมูลทั้งหมดจะส่งตรงไปแสดงในหน้าจอของแพทย์ทันที ดังรูปที่ 6



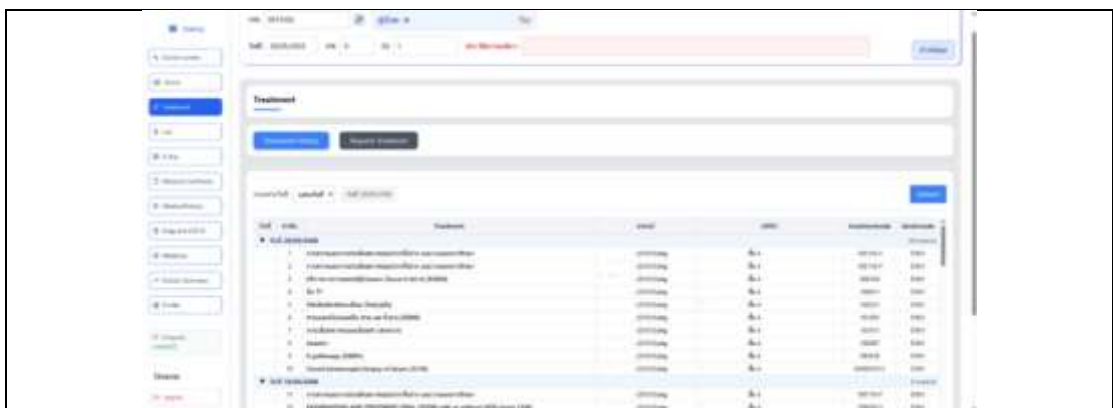
รูปที่ 6 ตัวอย่างหน้าจอการบันทึกข้อมูลพยาบาล

1.3 ส่วนงาน PTScreen คือหน้าจอสำหรับซักประวัติผู้ป่วยอย่างละเอียด รองรับ การแสดงผลและบันทึกข้อมูลได้หลายภาษา เช่น ไทย อังกฤษ พม่า และจีน ครอบคลุมการคัดกรอง ประวัติการแพ้ยา โรคประจำตัว ประวัติการผ่าตัด และพฤติกรรมสุขภาพ โดยมีระบบล็อกข้อมูลเพื่อป้องกันการแก้ไขซ้ำซ้อน ดังรูปที่ 7



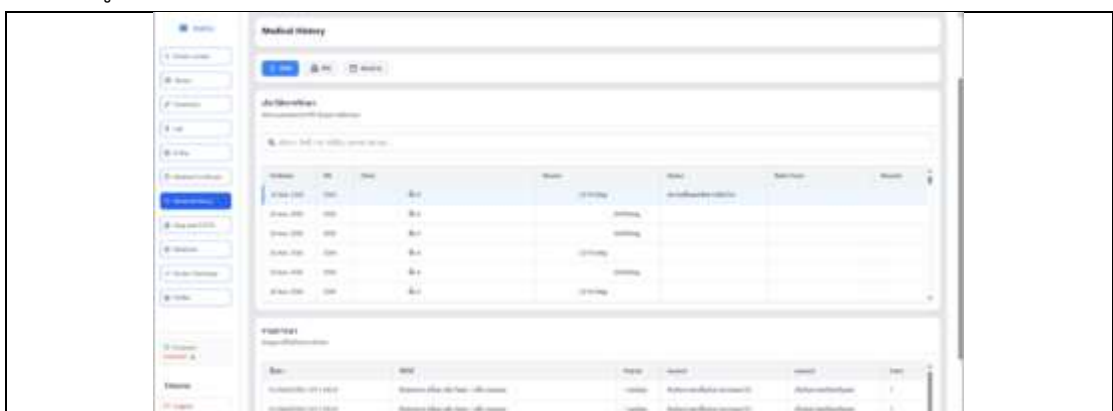
รูปที่ 7 ตัวอย่างหน้าจอการซึกประวัติผู้ป่วยนอก

1.4 ส่วนงาน Treatment คือหน้าจอสำหรับแพทย์และพยาบาลใช้สั่งและติดตามหัตถการทางการแพทย์ โดยระบบจะดึงข้อมูลสิทธิ์การรักษาและค่าใช้จ่ายพร้อมส่งคำสั่งหัตถการทุกรายการเข้าระบบ HIS โดยอัตโนมัติ ดังรูปที่ 8



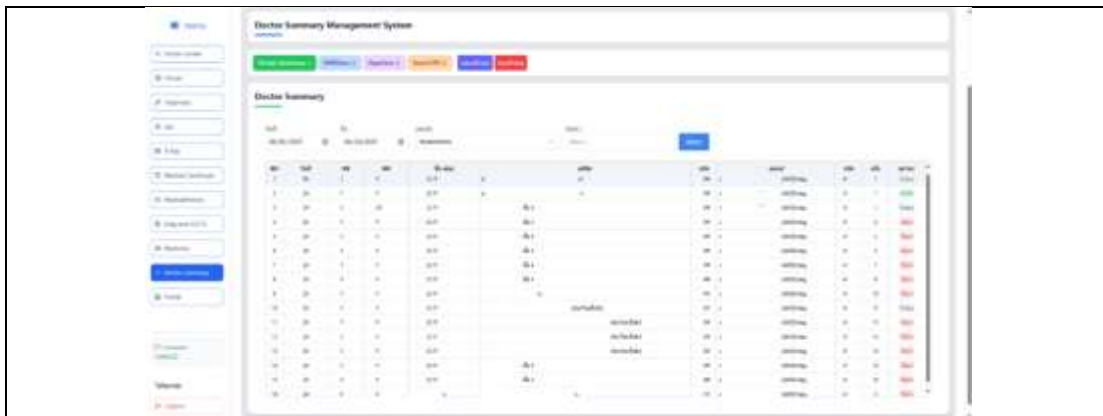
รูปที่ 8 ตัวอย่างหน้าจอการรักษา การบำบัด การดูแล การปฏิบัติ หรือวิธีการจัดการ

1.5 ส่วนงาน Medical History คือหน้าจอสำหรับดูประวัติการรักษาของผู้ป่วยย้อนหลัง โดยระบบจะดึงข้อมูลทั้งประวัติการรักษา การวินิจฉัยโรค ประวัติการใช้ยา และผลตรวจทางห้องปฏิบัติการมาแสดงผลรวมในหน้าเดียว แบบเรียลไทม์ เพื่อให้แพทย์เห็นภาพรวมของผู้ป่วยได้ทันที ดังรูปที่ 9



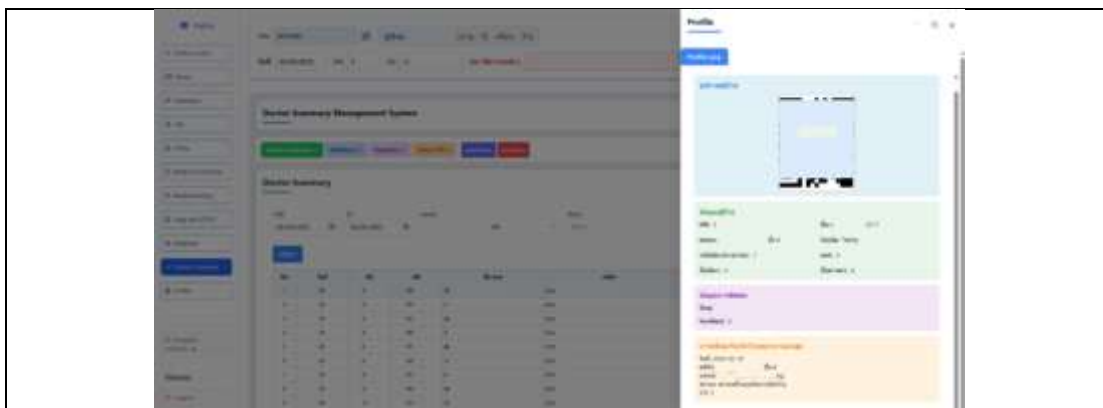
รูปที่ 9 ตัวอย่างหน้าจอแสดงประวัติการรักษาทั้งหมดของผู้ป่วย

1.6 ส่วนงาน Doctor Summary คือหน้าจอสำหรับแพทย์ติดตามและสรุปข้อมูลการรักษาของผู้ป่วยที่ตนเองดูแลตามช่วงวันที่ที่กำหนด สามารถค้นหากรองตามชื่อ หมายเลขผู้ป่วย หรือคลินิก รวมถึงสามารถคลิกเชื่อมโยงไปยังระบบภายนอกเช่น EMRDocs, Paperless ได้จากหน้าจอเดียว ดังรูปที่ 10



รูปที่ 10 ตัวอย่างหน้าจอการเอกสารสรุปข้อมูลประวัติการรักษาพยาบาล

1.7 ส่วนงาน Profile คือหน้าจอแสดงรายละเอียดข้อมูลส่วนตัวของผู้ป่วยอย่างครบถ้วน เช่น ชื่อ เพศ วันเกิด เลขบัตรประชาชน ที่อยู่ รูปภาพผู้ป่วย และประวัติการมาพบแพทย์ครั้งล่าสุด ดังรูปที่ 11



รูปที่ 11 ตัวอย่างหน้าจอการแสดงรายละเอียดข้อมูลผู้ป่วย

2. ผลการประเมินคุณภาพการทำงานของระบบสารสนเทศ

ผู้วิจัยได้ทดสอบการทำงานของแต่ละส่วนงานโดยใช้สถานการณ์จำลองทางคลินิก ร่วมกับชุดข้อมูลสมมติ จำนวน 110 กรณีทดสอบ ทำการทดสอบซ้ำจำนวน 3 รอบ เพื่อตรวจสอบและประเมินว่าระบบทำงานได้ถูกต้องตามเงื่อนไขที่ออกแบบไว้ โดยมีเกณฑ์การประเมินคือระบบต้องประมวลผลถูกต้องร้อยละ 100 ผลการทดสอบมีรายละเอียดดังนี้

ตารางที่ 1 ผลการทดสอบการทำงานตามหน้าที่ของระบบแยกตามส่วนงานหลัก

ส่วนงาน	รายการทดสอบหลัก	จำนวนกรณีทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
Doctor Screen	การบันทึกประวัติ การตรวจ CC, PI, PE, Doctor Note และระบบแจ้งเตือน	25	ระบบบันทึกและโหลดข้อมูลการตรวจได้ถูกต้องตามผู้ป่วยแต่ละราย	ผ่าน (100%)
Nurse Counter	การบันทึก Vital Signs และการส่งตัวผู้ป่วยเข้าห้องตรวจ	15	ระบบรับข้อมูลสัญญาณชีพตรวจสอบสิทธิ์และบันทึกสถานะห้องตรวจ	ผ่าน (100%)
PTScreen	การซักประวัติผู้ป่วยนอก (ประวัติโรค แพ้ยา การผ่าตัด บุหรี่/สุรา)	20	ระบบรับและบันทึกข้อมูลประวัติผู้ป่วยรองรับหลายภาษาได้ครบถ้วน	ผ่าน (100%)
Treatment	การค้นหาและส่งคำสั่งหัตถการ พร้อมยืนยันรายการก่อนบันทึก	15	ระบบดึงรายการหัตถการตามสิทธิ์และบันทึกคำสั่งลงระบบ HIS	ผ่าน (100%)
Medical History	การเรียกดูประวัติการรักษา OPD และรายการยาย้อนหลัง	15	ระบบแสดงประวัติการรักษาและรายละเอียดยาตามการมาตรวจแต่ละครั้ง	ผ่าน (100%)
Doctor Summary	การสรุปรายชื่อผู้ป่วยของแพทย์ตามช่วงวันที่ และการเชื่อมโยงระบบภายนอก	10	ระบบแสดงรายการผู้ป่วยพร้อมสถานะและเปิดลิงก์ระบบภายนอกได้ถูกต้อง	ผ่าน (100%)

ตารางที่ 1 ผลการทดสอบการทำงานตามหน้าที่ของระบบแยกตามส่วนงานหลัก (ต่อ)

ส่วนงาน	รายการทดสอบหลัก	จำนวนกรณีทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
Profile	การแสดงข้อมูลส่วนตัวผู้ป่วยและประวัติการมาโรงพยาบาลล่าสุด	10	ระบบดึงและแสดงข้อมูลส่วนตัวผู้ป่วยพร้อมรูปถ่ายได้ครบถูกต้อง	ผ่าน (100%)
รวม		110		ผ่าน (100%)

จากตารางที่ 1 พบว่าระบบมีความถูกต้องในเชิงหน้าที่ตามขอบเขตที่กำหนดไว้ โดยมีผลการทดสอบผ่านเกณฑ์ร้อยละ 100 ในทุกส่วนงาน ตัวอย่างกรณีทดสอบที่สำคัญ ได้แก่ การทดสอบระบบแจ้งเตือนการแพ้ยาในส่วนงาน Doctor Screen โดยจำลองสถานการณ์ให้ผู้ป่วยมีประวัติแพ้ยากลุ่ม Penicillin เมื่อแพทย์ทำการค้นหาและเลือกสั่งยาดังกล่าว ระบบสามารถตัดการทำงานและแสดงหน้าต่างแจ้งเตือนสีแดง ได้อย่างแม่นยำในทันที ซึ่งยืนยันได้ว่าระบบมีศักยภาพและความปลอดภัยเพียงพอในการนำไปประยุกต์ใช้จริงภายในงานคลินิกของหน่วยบริการผู้ป่วยนอก

สรุปผลการวิจัย

ผลการดำเนินงานวิจัยสามารถสรุปผลตามวัตถุประสงค์ได้ดังนี้ 1) ผลการพัฒนาระบบผู้วิจัยสามารถพัฒนาระบบสารสนเทศในโรงพยาบาลสำหรับงานผู้ป่วยนอก ในรูปแบบเว็บแอปพลิเคชันได้สำเร็จ โดยระบบสามารถเชื่อมโยงข้อมูลกับระบบ HIS เดิมผ่าน API และครอบคลุมการทำงานทั้ง 7 ส่วนงานหลักได้อย่างครบถ้วน และ 2) ผลการประเมินคุณภาพผลการทดสอบความถูกต้องเชิงฟังก์ชันจากกรณีทดสอบทั้งหมด 110 กรณี พบว่าระบบสามารถทำงาน ประมวลผล และแสดงผลการแจ้งเตือนต่างๆ ได้อย่างถูกต้องร้อยละ 100

อภิปรายผลการวิจัย

ผลการทดสอบความถูกต้องเชิงฟังก์ชันที่ผ่านเกณฑ์ร้อยละ 100 สะท้อนให้เห็นถึงประสิทธิภาพของสถาปัตยกรรมแบบ Three-tier Architecture ที่แบ่งแยกชั้นแสดงผล (React.js) ชั้นประมวลผล (Node.js) และชั้นจัดเก็บข้อมูล (MySQL) ออกจากกันอย่างเด็ดขาด การออกแบบนี้ช่วยให้การประมวลผลตรรกะทางคลินิกและการรับส่งข้อมูลผ่าน API ทำงานร่วมกับระบบสารสนเทศ

เดิม (Legacy HIS) ได้อย่างมีประสิทธิภาพและในรูปแบบเรียลไทม์ ซึ่งสอดคล้องกับแนวคิดของ Hoyt และ Hersh (2018) ที่ระบุว่าสถาปัตยกรรมสารสนเทศทางการแพทย์ที่ดีต้องรองรับการเข้าถึงข้อมูลที่ยืดหยุ่นและบำรุงรักษาง่าย นอกจากนี้ การพัฒนาระบบเป็นเว็บแอปพลิเคชันยังช่วยจัดข้อจำกัดของระบบเดสก์ท็อปเดิมได้อย่างเป็นรูปธรรม ได้แก่ 1) ลดภาระงานและต้นทุนด้วยการรวมศูนย์ข้อมูล ทำให้ดูแลและอัปเดตระบบได้จากเซิร์ฟเวอร์ส่วนกลาง 2) เพิ่มความยืดหยุ่นบุคลากรเข้าถึงข้อมูลได้จากทุกจุดบริการ (Abd Ghani & Jali, 2021) และ 3) ยกระดับความปลอดภัยด้วยระบบแจ้งเตือนอัตโนมัติ ซึ่งสอดคล้องกับงานวิจัยของ นิษฐา หอมวุฒิมังค์ และคณะ (2561) และโครงสร้างมาตรฐานข้อมูลด้านสุขภาพ (กองยุทธศาสตร์และแผนงาน สำนักงานปลัดกระทรวงสาธารณสุข, 2565) อย่างไรก็ตาม ระบบยังมีข้อจำกัดที่ต้องพึ่งพาเสถียรภาพของเครือข่ายเป็นหลัก

แนวทางการพัฒนาในอนาคต ผู้วิจัยมีแผนพัฒนาระบบจัดการข้อมูลแคชฝั่งไคลเอนต์ (Client-side Caching) เพื่อรองรับการทำงานโหนดออฟไลน์ชั่วคราว และขยายขอบเขตการทำงานไปยังส่วนงานผู้ป่วยใน (IPD) เพื่อให้ครอบคลุมกระบวนการบริหารทั้งองค์กรต่อไป

ข้อเสนอแนะในการวิจัย

ข้อเสนอแนะในการนำผลการวิจัยไปใช้ โรงพยาบาลควรเตรียมความพร้อมด้านระบบเครือข่ายอินเทอร์เน็ตให้เสถียรเพื่อรองรับการรับส่งข้อมูลแบบเรียลไทม์ ควบคู่ไปกับการจัดอบรมการใช้งานให้แก่บุคลากรทางการแพทย์ เพื่อให้การเปลี่ยนผ่านระบบเป็นไปอย่างมีประสิทธิภาพ

ข้อเสนอแนะในการวิจัยครั้งต่อไป ควรมีการวิจัยเพื่อขยายขอบเขตการทำงานของระบบให้ครอบคลุมแผนกอื่น ๆ และบูรณาการเทคโนโลยีปัญญาประดิษฐ์ เข้ามาช่วยวิเคราะห์ข้อมูล

เอกสารอ้างอิง

กองยุทธศาสตร์และแผนงาน สำนักงานปลัดกระทรวงสาธารณสุข. (2565). **โครงสร้างมาตรฐาน**

ข้อมูลด้านสุขภาพ กระทรวงสาธารณสุข. กระทรวงสาธารณสุข. <https://spd.moph.go.th/>
นิษฐา หอมวุฒิมังค์ และคณะ. (2561). การพัฒนาระบบคอมพิวเตอร์เพื่อแจ้งเตือนการแพ้ยา ร่วมกับ

การจัดการฐานข้อมูลการแพ้ยาของโรงพยาบาลแห่งหนึ่งในจังหวัดสมุทรสาคร. **วารสารเภสัชกรรมไทย**, 10(2), 433-444. [https://he01.tci-](https://he01.tci-thaijo.org/index.php/TJPP/article/view/171581)

[thaijo.org/index.php/TJPP/article/view/171581](https://he01.tci-thaijo.org/index.php/TJPP/article/view/171581)

ราชกิจจานุเบกษา. (2562, 27 พฤษภาคม). **พระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562.** เล่ม 136 ตอนที่ 69 ก, 52-95.

http://www.ratchakitcha.soc.go.th/DATA/PDF/2562/A/069/T_0052.PDF

สมาคมเวชสารสนเทศไทย. (2564). **กรอบการพัฒนาคุณภาพเทคโนโลยีสารสนเทศโรงพยาบาล (Hospital IT Quality Improvement Framework) (ฉบับปรับปรุง Version 2).**

https://tmi.or.th/wp-content/uploads/2021/12/HospitalIT_QualityImproveFramework_V2.pdf

Abd Ghani, M. K., & Jali, M. Z. (2021). Web-based hospital information system (HIS) for outpatient management. **International Journal of Advanced Computer Science and Applications**, 12(4), 112-118.

<https://thesai.org/Publications/IJACSA>

Federal Office for Information Security (BSI). (2025). **Guidance for secure Hospital Information Systems (HIS)**.

https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/DigitaleGesellschaft/SiKIS_Guidance.pdf

Health Insurance Portability and Accountability Act of 1996, Pub. L. No. 104-191, **110 Stat. 1936** (1996). <https://www.govinfo.gov/app/details/PLAW-104publ191>

Hoyt, R. E., & Hersh, W. R. (2018). **Health informatics: Practical guide for healthcare and information technology professionals** (7th ed.). Informatics Education.

<https://dmice.ohsu.edu/hersh/informaticsbook/>

IBM. (2024). **What is three-tier architecture?**. IBM Think.

<https://www.ibm.com/think/topics/three-tier-architecture>

ITU Online. (2024, March 29). **What is three-tier architecture?**. ITU Online IT Training. <https://www.ituonline.com/tech-definitions/what-is-three-tier-architecture/>

Kuo, M. H., Kushniruk, A. W., & Borycki, E. M. (2020). Design and implementation of a clinical information system for outpatient care. **Journal of Medical Systems**, 44(2), 55-62. <https://link.springer.com/journal/10916>

Nelson, R., & Staggers, N. (2018). **Health informatics: An interprofessional approach** (2nd ed.). Elsevier. <https://www.elsevier.com/books/health-informatics/nelson/978-0-323-40231-6>