

# Traceability assessment model for checking traceability requirement compliance of a medical device software development process with ISO/IEC 62304

**Songsakdi Rongviriyapanish, Natsuda Kasisopha**

Computer science department, Thammasat university

---

## **Abstract**

Medical device software must take the safety of the user into serious consideration since the failure of the medical device software can lead to hazardous situations in the user's life. The software development process could ensure the user's safety by the software system problem resolution's traceability. The traceability should start from the beginning of the medical device software development process to the maintenance lifecycle. This demonstrates that the safe software has been built using a defined process. This document describes the ISO/IEC62304 Medical device software standard, software development

lifecycle, and traceability in the process of solving problems. This approach has focused on small and medium organizations.

**Keywords:** Medical Device Software Traceability, Medical Device Software Process Assessment, Medical Device Software Standard

*Received: 10 January 2023, Revised: 25 March 2023, Accepted: 1 May 2023*

---

**Correspondence:** Songsak Rongviriyapanich, Computer science department, Thammasat university, 99 Moo 18 Khlong Nueng, Khlong Luang District, Pathum Thani 12120, E-mail: songsak\_r@sci.tu.ac.t

# โมเดลสำหรับประเมินกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เกี่ยวกับข้อกำหนดด้านการตรวจสอบย้อนกลับของกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 62304

**รศ.ศักดิ์ รongviriyapanish, นัฐสุดา กสิโสภ**

สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

## บทคัดย่อ

ซอฟต์แวร์อุปกรณ์การแพทย์ต้องคำนึงถึงความปลอดภัยของผู้ใช้เป็นสำคัญ เนื่องจากความล้มเหลวของซอฟต์แวร์อุปกรณ์การแพทย์สามารถนำไปสู่ผลลัพธ์ที่ร้ายแรงถึงชีวิตผู้ใช้ได้ การแบ่งกลุ่มความปลอดภัยของซอฟต์แวร์อุปกรณ์การแพทย์และการจัดการความเสี่ยงจึงมีบทบาทสำคัญในการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เพื่อให้แน่ใจว่าซอฟต์แวร์ที่ปลอดภัยได้รับการพัฒนาอย่างเหมาะสม การแสดงความสามารถในการตรวจสอบย้อนกลับของความต้องการได้ตลอดวงจรชีวิตการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์จนถึงการบำรุงรักษาซอฟต์แวร์ เป็นส่วนสำคัญในการแสดงให้เห็นว่าซอฟต์แวร์ที่ปลอดภัย ได้ถูกสร้างขึ้นโดยใช้กระบวนการที่กำหนดไว้ ดังนั้นมาตรฐานและแนวทางการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์จึงเน้นย้ำถึงความจำเป็นในการตรวจสอบย้อนกลับ เอกสารนี้

ได้อธิบายถึงมาตรฐาน ISO/IEC62304 Medical device software วงจรชีวิตการพัฒนาซอฟต์แวร์ และการตรวจสอบย้อนกลับในกระบวนการการแก้ปัญหาเมื่อมีเหตุที่อุปกรณ์การแพทย์ไม่สามารถใช้งานได้ สำหรับนำไปใช้ในองค์กรขนาดเล็กและขนาดกลาง

**คำสำคัญ:** การตรวจสอบย้อนกลับซอฟต์แวร์อุปกรณ์การแพทย์, การประเมินกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์, มาตรฐานซอฟต์แวร์อุปกรณ์การแพทย์

วันที่รับต้นฉบับ: 10 มกราคม 2566, วันที่แก้ไข: 25 มีนาคม 2566, วันที่ตอบรับ: 1 พฤษภาคม 2566

## บทนำ

### 1. เกริ่นนำ และที่มาของปัญหา

ซอฟต์แวร์สำหรับอุปกรณ์การแพทย์มีความจำเป็นต้องผ่านการตรวจสอบมาตรฐานที่กำหนดโดยกองควบคุมเครื่องมือแพทย์ภายใต้สำนักงานอาหารและยา (อย.) [9] อาทิ การประเมินความเสี่ยง การประเมินมาตรฐาน ISO/IEC62304 Medical device software — Software life cycle processes [5] ก่อนยื่นขออนุญาตจัดแจ้งผลิตภัณฑ์ ซึ่งตามมาตรฐานกำหนดให้ผู้ผลิตซอฟต์แวร์สำหรับอุปกรณ์การแพทย์จำเป็นต้องแสดงความสัมพันธ์เชื่อมโยงจากข้อกำหนดซอฟต์แวร์ (Software Requirement Specification) ไปถึงขั้นตอนการวางตลาด ขั้นตอนการบำรุงรักษาซอฟต์แวร์และการรับแจ้งคำติชมจากผู้ใช้ การเช็คความสัมพันธ์ดังกล่าวมีความครบถ้วนสมบูรณ์ตามมาตรฐานถูกเรียกว่า การตรวจสอบย้อนกลับการพัฒนาซอฟต์แวร์ (Software Traceability) โดยผู้ผลิตต้องแสดง

ความสัมพันธ์เชื่อมโยงอย่างชัดเจนผ่านแนวทางปฏิบัติ (Standard Operating Procedure (SOP)) แผนงานการปฏิบัติ (Plan) บันทึก (Record) และเอกสารรายงาน (Document)

แม้ว่าในมาตรฐานย่อยระบุให้ผู้ผลิตซอฟต์แวร์สำหรับอุปกรณ์การแพทย์ต้องแสดงการตรวจสอบย้อนกลับการพัฒนาซอฟต์แวร์ แต่ไม่ได้ระบุวิธีการว่าต้องทำอะไรเพื่อให้เป็นไปตามข้อกำหนดตามมาตรฐาน ISO/IEC 62304 ซึ่งทำให้บริษัทผู้พัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เกิดความสับสน เสียเวลาในการทำความเข้าใจ เสียโอกาสต่อการตลาด และเกิดค่าใช้จ่ายเพิ่มขึ้นเพื่อให้ผ่านมาตรฐาน อาทิเช่น จ้างที่ปรึกษา หรือซื้อชุดเครื่องมือช่วยตรวจสอบมาตรฐาน

จากประสบการณ์ของผู้วิจัยจากการเป็นผู้ประเมินมาตรฐานการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ ภายใต้หน่วยงานงานวิศวกรรมซอฟต์แวร์และทดสอบผลิตภัณฑ์ (SEPT) ฝ่ายสนับสนุนบริการทางวิศวกรรมและเทคโนโลยี (TSS) ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) พบว่าผู้ผลิตซอฟต์แวร์อุปกรณ์การแพทย์ในประเทศไทยมักใช้โมเดลกระบวนการพัฒนามตามแบบจำลองวี (V-Model)

**ผู้นิพนธ์ประสานงาน:** ปานใจ ธารทัศน์วงศ์, ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์, มหาวิทยาลัยศิลปากร วิทยาเขต พระราชวังสนามจันทร์ ถนนราชมรรคาใน อาเภอเมือง จังหวัดนครปฐม 73000 E-mail: panjai@gmail.com

วัตถุประสงค์ของงานวิจัยนี้เพื่อนำเสนอโมเดลที่ช่วยสร้างความเข้าใจ (Guideline) ข้อกำหนดที่ผู้ผลิตซอฟต์แวร์อุปกรณ์การแพทย์ซึ่งพัฒนาซอฟต์แวร์ตามแบบจำลองวี (V-Model) ต้องดำเนินการเพื่อให้ผ่านข้อกำหนดการตรวจสอบย้อนกลับการพัฒนาซอฟต์แวร์ (Software Traceability) ตามมาตรฐาน ISO/IEC 62304 โดยหัวข้อที่ 2 อธิบายมาตรฐาน ISO/IEC 62304 สำหรับการพัฒนาซอฟต์แวร์เครื่องมือแพทย์ งานวิจัยที่ระบุเหตุผลที่กระบวนการพัฒนาเอาใจใส่ไม่ได้รับความนิยมนเท่ากับแบบจำลองวี งานวิจัยที่เกี่ยวข้องกับการแนะนำแนวทางสำหรับประเมินการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ให้สอดคล้องกับมาตรฐาน ส่วนที่ 3 ผู้วิจัย อธิบายแนวทางการสร้างโมเดลสำหรับประเมินกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เทียบกับข้อกำหนดตามมาตรฐาน ISO/IEC 62304 หัวข้อที่ 4 แสดงตัวอย่างโมเดลสำหรับประเมินกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ให้สอดคล้องกับข้อกำหนดการตรวจสอบย้อนกลับการพัฒนาซอฟต์แวร์ ส่วนที่ 9 (Clause 9 Software Problem Resolution) ตามมาตรฐาน ISO/IEC 62304

**2. มาตรฐานและงานวิจัยที่เกี่ยวข้อง**

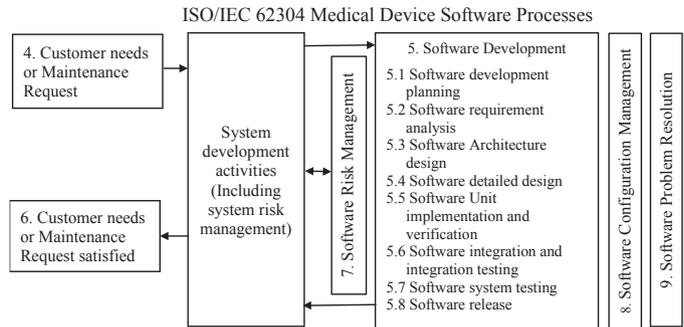
**A. ISO/IEC 62304 Medical device software — Software life cycle processes**

ตามมาตรฐานอุตสาหกรรมเครื่องมือแพทย์ที่กำหนดโดยสำนักงานคณะกรรมการอาหารและยา (อย.) ระบุให้ผู้ผลิตซอฟต์แวร์อุปกรณ์การแพทย์ต้องจัดทำเอกสารที่สอดคล้องกับมาตรฐาน ISO/IEC 62304 ซึ่งเกี่ยวข้องกับขั้นตอนการจัดการความต้องการที่เกี่ยวข้องกับซอฟต์แวร์อุปกรณ์การแพทย์ ขั้นตอนที่เกี่ยวข้องกับการพัฒนาระบบซึ่งรวมฮาร์ดแวร์และซอฟต์แวร์ ขั้นตอนการประเมินความเสี่ยงของระบบซอฟต์แวร์ ขั้นตอนการพัฒนาซอฟต์แวร์ ขั้นตอนการบริหารจัดการ ขั้นตอนการแก้ไขปัญหา ขั้นตอนการบำรุงรักษาซอฟต์แวร์ ดังรูปที่ 1

มาตรฐาน ISO/IEC 62304 ถูกแบ่งเนื้อหาเป็น 9 ส่วน ส่วนที่ 4-9 เป็นข้อกำหนดที่เกี่ยวข้องกับกระบวนการพัฒนาซอฟต์แวร์ที่ผู้ผลิตซอฟต์แวร์อุปกรณ์การแพทย์ต้องปฏิบัติตาม

ส่วนที่ 4 กล่าวถึงข้อกำหนดทั่วไปสำหรับผู้ผลิตอุปกรณ์ทางการแพทย์ (General Requirement) เช่น ความสามารถในการจัดการการบริหารจัดการความต้องการของลูกค้า มีระบบการจัดการด้านคุณภาพที่อ้างอิงตาม ISO 13485 เพื่อผลิตหรือจัดหาซอฟต์แวร์ให้ได้ตามความต้องการและวัตถุประสงค์ กำหนดให้มีระบบการจัดการด้านความเสี่ยงอ้างอิงตาม ISO 14971 และต้องสอดคล้องกับมาตรฐาน ISO/IEC 62304 ส่วนที่ 7 กระบวนการจัดการความเสี่ยงด้านซอฟต์แวร์ (Software Risk Management Process) ต้องมีการจัดทำเป็นเอกสารระบุแนวทางปฏิบัติ แผนงานเพื่อพัฒนาซอฟต์แวร์ให้ได้ตรงตามความต้องการและตรงตามวัตถุประสงค์ของระบบ ในส่วนที่ 4 นี้ยังอธิบายการจำแนกซอฟต์แวร์อุปกรณ์การแพทย์ออกเป็น 3 คลาส แยกตามระดับความปลอดภัยของซอฟต์แวร์ (Software safety class) ขึ้นอยู่กับความ

รุนแรง (Severity) ที่ซอฟต์แวร์อาจทำให้เกิด ดังนี้ Class A: ซอฟต์แวร์ที่ไม่ทำให้ผู้ใช้งานบาดเจ็บหรือเกิดผลเสียต่อสุขภาพ Class B: ซอฟต์แวร์ที่อาจทำให้เกิดการบาดเจ็บเล็กน้อย (Non-Serious Injury) และ Class C: ซอฟต์แวร์ที่อาจทำให้เกิดการบาดเจ็บขั้นสาหัส (Serious Injury) หรือตายได้ [8]



รูปที่ 1 ข้อกำหนดด้านกระบวนการพัฒนาซอฟต์แวร์ อุปกรณ์การแพทย์ตามมาตรฐาน ISO/IEC 62304

ส่วนที่ 5 อธิบายถึงกระบวนการพัฒนาซอฟต์แวร์ (Software Development Process) ซึ่งมีขั้นตอนย่อยดังนี้ (1) การวางแผนพัฒนาซอฟต์แวร์ ซึ่งเป็นการนำเสนอขอบเขตของโครงการซอฟต์แวร์ (2) การวิเคราะห์ข้อกำหนดของซอฟต์แวร์ เป็นกำหนดหน้าที่การทำงาน (function) ตามความต้องการให้แก่ระบบย่อยเพื่อให้ระบบซอฟต์แวร์ทำงานตามข้อกำหนดของซอฟต์แวร์ และระบบตอบสนองความต้องการของลูกค้าหรือตามวัตถุประสงค์ (3) การออกแบบสถาปัตยกรรมซอฟต์แวร์เพื่อให้ตรงตามผลการวิเคราะห์จาก (2) และระบุวงจรชีวิตการพัฒนาซอฟต์แวร์ที่จะใช้ในโครงการ (4) การออกแบบรายละเอียดการทำงานของซอฟต์แวร์ คือการออกแบบหน่วยย่อยต่างๆของซอฟต์แวร์จากข้อมูลที่ได้จากขั้นตอนก่อนหน้านี้อและซอฟต์แวร์ตัวอื่นๆที่นำมาใช้งานร่วมกับกับหน่วยย่อย (5) การพัฒนาซอฟต์แวร์หน่วยย่อย หมายถึงการเขียนโค้ดเพื่อสร้างและตรวจสอบความถูกต้องซอฟต์แวร์หน่วยย่อย (6) การรวมซอฟต์แวร์และการทดสอบการรวมซอฟต์แวร์หน่วยย่อยให้เป็นระบบซอฟต์แวร์ (7) การทดสอบระบบซอฟต์แวร์ เป็นการทดสอบทั้งระบบให้ตรงกับข้อกำหนด (8) การเปิดตัวซอฟต์แวร์ หมายถึงการวางตลาดและปรับใช้ซอฟต์แวร์ ในส่วนที่ 5 นี้อ้างอิงตามมาตรฐาน ISO/IEC/IEEE 12207 Systems and Software Engineering — Software life cycle processes.

ส่วนที่ 6 คือกระบวนการบำรุงรักษาซอฟต์แวร์ (Software Maintenance Process) กล่าวถึงขั้นตอนการบำรุงรักษา เอกสารที่ต้องกรอกสำหรับกระบวนการนี้ รวมถึงการแก้ไขข้อบกพร่องของซอฟต์แวร์ที่พบระหว่างการบริหารรักษาและการประเมินความเสี่ยงด้านความปลอดภัยจากข้อบกพร่องที่เกิดขึ้น ข้อมูลจากส่วนนี้จัดทำเป็นเอกสารโดยระบุขั้นตอนปฏิบัติ แผนงาน และบันทึกผลการบำรุงรักษาซอฟต์แวร์ตามแผนงานที่กำหนด

ส่วนที่ 7 กระบวนการจัดการความเสี่ยงด้านซอฟต์แวร์ (Software Risk Management Process) หรือที่เรียกกันสั้นๆว่าการประเมินความเสี่ยง ส่วนนี้ทำหน้าที่วิเคราะห์ ประเมินความล้มเหลวของซอฟต์แวร์ที่อาจเกิดขึ้น ตลอดจนการจัดการคุณลักษณะด้านความปลอดภัยของซอฟต์แวร์ ซึ่งทำหน้าที่เป็นตัวควบคุมความเสี่ยงสำหรับความล้มเหลวของฮาร์ดแวร์ ข้อผิดพลาดในการใช้งาน และสำหรับความล้มเหลวของระบบอื่นๆ ในกระบวนการนี้ผู้ผลิตต้องระบุถึงอันตรายที่อาจเกิดขึ้นกับผู้ใช้ อาทิเช่น การบาดเจ็บทางร่างกาย ความเสียหาย หรือทั้งต่อสุขภาพของคนหรือความเสียหายต่อทรัพย์สินหรือสิ่งแวดล้อม แหล่งที่อาจเกิดอันตราย และความน่าจะเป็นที่จะเกิดอันตราย และความรุนแรงของอันตรายนั้นๆ เอกสารการประเมินความเสี่ยงของระบบซอฟต์แวร์ตามที่กำหนดในส่วนต่างๆ ของมาตรฐาน อาทิ ส่วนที่ 4 รวมแนวทางปฏิบัติในการประเมินความเสี่ยง

ส่วนที่ 8 กระบวนการบริหารการจัดเก็บผลิตภัณฑ์ที่ส่งมอบ (Software Configuration Management Process) ในการเขียนโค้ดของซอฟต์แวร์จำเป็นต้องมีระบบควบคุมซอร์สโค้ด สภาพแวดล้อมการพัฒนาและอภิยาวิธีจัดการการสร้าง (Source code build) การรวบรวมซอร์สโค้ด (Source code compile) และการจัดเก็บโค้ด รวมถึงการกำหนดกระบวนการเปลี่ยนแปลงซอฟต์แวร์ (Software Change Process) การควบคุมเวอร์ชันของระบบซอฟต์แวร์ (Version Control) อาทิ การใช้โปรแกรม Subversion (SVN) ในการจัดการกับเอกสาร ซอร์สโค้ด ไฟล์รูปภาพ ฯลฯ โปรแกรมจะทำหน้าที่จัดการควบคุมเวอร์ชันของไฟล์ต่างๆ ส่วนนี้ควรมีเอกสารระบุแนวทางปฏิบัติในการจัดการการจัดเก็บผลิตภัณฑ์ต่างๆ การบันทึกการเปลี่ยนแปลง การจัดการควบคุมเวอร์ชัน

ส่วนที่ 9 กระบวนการแก้ไขปัญหาซอฟต์แวร์ (Software Problem Resolution Process) ผู้ผลิตอุปกรณ์การแพทย์ต้องมีแนวทางและเตรียมแผนการรายงานปัญหา (9.1 Prepare problem reports) เมื่อได้รับแจ้ง การตรวจสอบปัญหาที่เกิดขึ้นเพื่อหาสาเหตุอันตราย (Harm) ความเสี่ยง (Hazard) สถานะการณ์ที่เสี่ยง (Hazardous situation) และระดับความรุนแรง (Measure) ของอันตราย ประเมินความเสี่ยงและโอกาสที่อาจเกิดขึ้น (9.2 Investigate the problem) หลังจากการหาสาเหตุของปัญหาและแนวทางแก้ไข ต้องดำเนินการแจ้งผู้ที่เกี่ยวข้อง อาทิเช่น ผู้ใช้หรือตัวแทนจำหน่าย (9.3 Advise relevant parties) เมื่อแนวทางแก้ไขได้รับความเห็นชอบในที่ประชุม ดำเนินการแก้ไข (9.4 Use change control process) ซึ่งกระบวนการควบคุมการเปลี่ยนแปลง จะดำเนินการตามคำร้องขอการเปลี่ยนแปลง (Change request) ผลการวิเคราะห์ปัญหา รวมถึงผลการอนุมัติคำร้องขอเปลี่ยนแปลง ทุกกิจกรรมที่ดำเนินงานต้องทำการบันทึกสถานะ (9.5 Maintain records) รวมถึงการวิเคราะห์แนวโน้มของปัญหาที่อาจเกิดขึ้นภายหลัง (9.6 Analyze problems for trends) การทวนสอบการแก้ปัญหาซอฟต์แวร์ (9.7 Verify software problem resolution) เป็นการทวนสอบการแก้ปัญหาและยืนยันความสามารถ

ในการทำงานของซอฟต์แวร์ยังคงทำได้ตามวัตถุประสงค์เดิมของระบบซอฟต์แวร์อุปกรณ์การแพทย์ ปัญหาได้รับการแก้ไขและปิดรายงานปัญหาแล้วหรือไม่ รวมถึงการตรวจสอบแนวโน้มว่ามีสิ่งที่ไม่พึงประสงค์เกิดขึ้นใหม่หรือไม่ (9.8 Test documentation contents) การทดสอบระบบหลังมีการแก้ไข ต้องจัดทำบันทึกจากกิจกรรมทวนสอบ แก้ไข และการทดสอบถดถอย (Regression Test) ฯลฯ เป็นต้น

F. McCaffery, et.al. [1] ได้จัดทำกรอบการประเมินกระบวนการชื่อ MDevSPICE แนะนำแนวทางสำหรับประเมินการพัฒนาซอฟต์แวร์อุปกรณ์ทางการแพทย์ที่พัฒนาตามมาตรฐาน ISO/IEC 15504 (SPICE) [6] ว่าเป็นไปตามข้อกำหนดมาตรฐานการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ ISO/IEC 12207:2008 และ IEC 62304 หรือไม่

B. วงจรชีวิตการพัฒนาซอฟต์แวร์ (Software Development Life Cycle (SDLC))

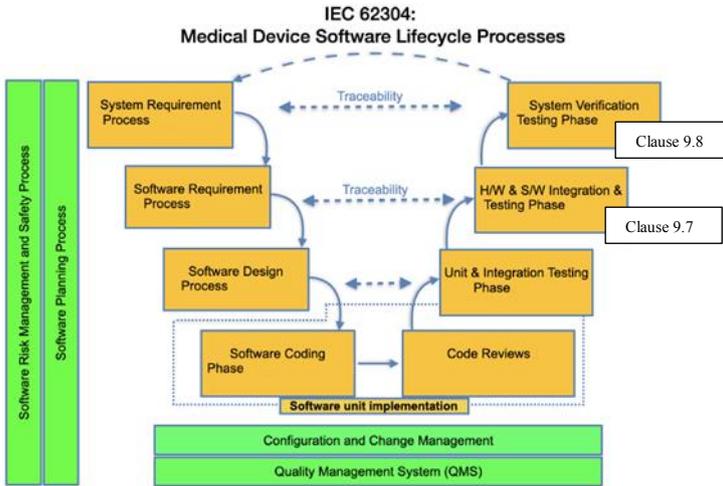
ในส่วนที่ 5 ตามมาตรฐาน ISO/IEC62304 กำหนดให้กระบวนการพัฒนาซอฟต์แวร์ของผู้พัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ต้องเป็นไปตามมาตรฐาน ISO/IEC/IEEE12207 ที่กำหนดมาตรฐานกระบวนการพัฒนาซอฟต์แวร์ โดยบริษัทผู้พัฒนาซอฟต์แวร์จำเป็นต้องเลือกใช้วงจรชีวิตการพัฒนาซอฟต์แวร์ (Software Development Life Cycle (SDLC)) เพื่อกำหนดขั้นตอนการพัฒนาซอฟต์แวร์ วงจรชีวิตการพัฒนาซอฟต์แวร์มีหลายรูปแบบ อาทิ โครงสร้างแบบน้ำตก (Waterfall) แบบจำลองวี (V Model) กระบวนการพัฒนาซอฟต์แวร์แบบอจิล (Agile Process)

M. Hugh, et.al. [2] ได้ทำการวิจัยสำรวจพบอุปสรรคที่ทำให้การนำกระบวนการการพัฒนาซอฟต์แวร์แบบคล่องตัวหรืออจิลมาใช้กับการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ เนื่องจากกระบวนการไม่สอดคล้องกับกฎระเบียบของสำนักงานอาหารและยาที่กำหนดไว้ของประเทศต่างๆ ตัวอย่างเช่น การจัดทำเอกสาร ซึ่งกระบวนการเอจายล์มักไม่เน้นการจัดทำเอกสาร ในขณะที่กฎระเบียบของสำนักงานอาหารและยา (FDA) กำหนดให้ต้องทำการตรวจสอบย้อนกลับระหว่างความต้องการกับทุกขั้นตอนการพัฒนาซอฟต์แวร์ หรือตามข้อกำหนดตามกฎระเบียบของสำนักงานอาหารและยา (FDA General Principles of Software Validation (GPSV)) ระบุให้ต้องจัดทำเอกสารความต้องการให้ชัดเจนก่อนการพัฒนาและการทดสอบ

N.M.A. Munassar, et.al. [3] ได้สำรวจพบว่าบริษัทผู้พัฒนาซอฟต์แวร์อุปกรณ์การแพทย์มักจะใช้กระบวนการตามแบบจำลองวี (V Model) เนื่องจากเป็นกระบวนการที่ผลิตเอกสารและผลลัพธ์ที่สอดคล้องกับข้อกำหนดตามมาตรฐานซอฟต์แวร์อุปกรณ์การแพทย์ โดยจากผลการสำรวจในงาน [4] พบว่า 50% ของกลุ่มสำรวจที่เป็นองค์กรที่พัฒนาซอฟต์แวร์ประเภท Safety Critical มีการใช้กระบวนการพัฒนาตามแบบจำลองวี 25% ใช้กระบวนการพัฒนาแบบอจิล และอีก 25% ใช้กระบวนการพัฒนาแบบอื่น ซึ่งซอฟต์แวร์อุปกรณ์การแพทย์จัดอยู่ในกลุ่มซอฟต์แวร์ประเภท Safety Critical

### 3. แนวทางการสร้างโมเดลสำหรับประเมินกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เทียบกับข้อกำหนดตามมาตรฐาน ISO/IEC 62304

#### A. กระบวนการพัฒนาซอฟต์แวร์ตามแบบจำลองวี (V Model) และมาตรฐาน ISO/IEC 62304



รูปที่ 2 กระบวนการพัฒนาซอฟต์แวร์ตามแบบจำลองวีเพิ่มเติมด้วยกระบวนการที่จำเป็นตามมาตรฐาน ISO/IEC 62304 [7]

รูปที่ 2 แสดงขั้นตอนหลักในกระบวนการพัฒนาตามแบบจำลองวี โดยการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ตามกระบวนการแบบจำลองวีเริ่มจากการกำหนดความต้องการของระบบโดยภาพรวมและค่อยลงลึกไปถึงการกำหนดความต้องการเฉพาะส่วนซอฟต์แวร์และการออกแบบ การเขียนโค้ด ตามลำดับแบบจำลองวี (V Model) เป็นโมเดลที่เน้นการตรวจสอบ (verification) และการรับรองความถูกต้อง (validation) โดยสามารถทำการทดสอบผลิตภัณฑ์ซอฟต์แวร์ที่ขนานกันไปกับการพัฒนาซอฟต์แวร์ มีการทวนสอบซอฟต์แวร์ (Software verification) และทดสอบความถูกต้องทุกขั้นตอน ซึ่งช่วยสร้างความมั่นใจในการพัฒนาซอฟต์แวร์ว่าซอฟต์แวร์จะทำงานตามที่ผู้กำหนด

ผลจากการดำเนินการตามกระบวนการตามแบบจำลองวี ส่งผลให้กระบวนการพัฒนาซอฟต์แวร์เป็นไปตามข้อกำหนดตามมาตรฐาน ISO/IEC 62304 บางส่วน เช่น ขั้นตอนการทดสอบตรวจทานระบบ (System Verification Testing Phase) สอดคล้องกับข้อกำหนดส่วน 9.7 และ ขั้นตอนทดสอบการเชื่อมต่อระบบส่วนฮาร์ดแวร์และซอฟต์แวร์ (HW & SW Integration and Test Process) จะทำให้มีการสร้างเอกสารการทดสอบ (Test documents) ที่มีคุณสมบัติครบตามมาตรฐานส่วน 9.8 อย่างไรก็ตาม นอกจากการดำเนินการตามขั้นตอนหลักตามแบบจำลองวีแล้ว ผู้พัฒนาซอฟต์แวร์อุปกรณ์การแพทย์จำเป็นต้องทำกระบวนการเสริมเพื่อให้ผ่านมาตรฐาน ISO/IEC 62304 ได้แก่ กระบวนการจัดการความเสี่ยง

และความปลอดภัยจากการใช้ซอฟต์แวร์ (Software Risk Management and Safety Process) กระบวนการวางแผนพัฒนาซอฟต์แวร์ (Software Planning Process) กระบวนการบริหารการจัดเก็บผลิตภัณฑ์ที่ส่งมอบและจัดการความเปลี่ยนแปลง (Configuration and Change Management) และ กระบวนการระบบจัดการคุณภาพ (Quality Management System) อ้างอิง [7]

#### B. เกณฑ์การตรวจสอบย้อนกลับของมาตรฐาน

งานวิจัยนี้เสนอโมเดลที่แสดงข้อกำหนดเพื่อใช้ในการตรวจสอบย้อนกลับกระบวนการพัฒนาเทียบกับข้อกำหนดมาตรฐาน ISO/IEC 62304 ในรูปแบบตารางที่ประกอบด้วย คอลัมน์ 6 คอลัมน์ คือ 1) คอลัมน์ Clause ระบุชื่อหมายเลขส่วนย่อยในมาตรฐาน 2) คอลัมน์ Description ที่อธิบายสาระสำคัญกำหนดในมาตรฐานแต่ละส่วน (Clause) 3) คอลัมน์ Standard Operating Procedure (SOP) ที่อธิบายกระบวนการที่ต้องมีการดำเนินการเพื่อผ่านมาตรฐานในแต่ละส่วนย่อย 4) คอลัมน์ Plan ที่อธิบายแผนที่จำเป็นต้องมี 5) คอลัมน์ Record ที่กำหนดข้อมูลที่ต้องถูกบันทึกเป็นรายการหลักฐาน 6) คอลัมน์ Document ที่กำหนดเอกสารที่จำเป็นต้องมี ตารางที่ 1 แสดงตัวอย่างการอธิบายข้อกำหนด SOP, Plan, Record และ Document สำหรับมาตรฐานส่วนที่ 9 ของ ISO/IEC 62304

ในการตรวจสอบย้อนกลับของมาตรฐาน ISO/IEC 62304 ต้องมีเกณฑ์การวัดเพื่อให้เห็นความเชื่อมโยงระหว่างมาตรฐานกับขั้นตอนการดำเนินการ (SOP) แผน (Plan) รายการข้อมูลที่มีการบันทึกเป็นหลักฐาน (Record) และเอกสารที่ผู้ผลิตจัดทำขึ้น (Document) ข้อมูลทั้ง 4 ส่วนจะถูกนำมาเปรียบเทียบตามเกณฑ์การทำ Internal Audits ตามมาตรฐาน ISO 13485:2016 ซึ่งมีเกณฑ์ดังนี้ ก) การมีแนวทางปฏิบัติและคำอธิบาย ข) ตรวจสอบประสิทธิภาพของการดำเนินการตามแนวทางปฏิบัติ ค) จัดทำเอกสารนโยบาย โปรโตคอล และบันทึกการตรวจสอบภายในทั้งหมดอย่างละเอียดถี่ถ้วน ง) ประเมินความสอดคล้องระหว่างเอกสารและมาตรฐาน ISO/IEC 62304 เพื่อที่จะทำการแก้ไขส่วนที่ขาดหายไม่ครบตามมาตรฐาน

ยกตัวอย่างการตรวจสอบย้อนกลับของข้อ 9.7 กระบวนการทวนสอบการแก้ปัญหา ตามตารางที่ 3 ในการตรวจสอบย้อนกลับผู้พัฒนาต้องมีภาระและอธิบายกระบวนการ 2 กระบวนการ คือ กระบวนการแก้ไขปัญหาที่เกิดจากการใช้ซอฟต์แวร์ (Software Problem Resolution) และ กระบวนการจัดการการเปลี่ยนแปลง (Change Management) ตามเกณฑ์ข้อ ก) และกระบวนการทั้งสองต้องมีขั้นตอนที่สามารถนำมาปฏิบัติได้จริงตามเกณฑ์ข้อ ข) การจัดทำเอกสารตามเกณฑ์ข้อ ค) คือ ต้องมีการจัดทำแผนการบำรุงรักษาซอฟต์แวร์ (Software Maintenance Plan) มีการบันทึกรายละเอียดปัญหาที่เกิดขึ้นเป็นรายการหลักฐาน (Problem Record) มีรายการผลการทดสอบ มีรายการอนุมัติการแก้ไข มีการจัดทำรายงานการแก้ไขปัญหา (Problem Report) มีแบบฟอร์มติดตาม

สถานการณ์แก้ไขปัญหาเกณฑ์ข้อ ง) ประเมินว่าเนื้อหาที่กรอกในแบบฟอร์มการแจ้งปัญหา รายงานการแก้ไขปัญหา และเอกสารรายการข้อมูลต่างๆมีเนื้อหาที่สอดคล้องกับกระบวนการที่ระบุไว้ ขั้นตอนและตามที่กำหนดในมาตรฐานส่วนย่อยที่ 9.7 ตามเกณฑ์การประเมินตารางที่ 1 ตารางที่ 2 แสดงตัวอย่างขั้นตอนการดำเนินการเพื่อแก้ไขปัญหาที่เกิดจากการใช้ซอฟต์แวร์ (SOP of Software Problem Resolution)

**unaşu**

ก่อนการวางตลาดของซอฟต์แวร์อุปกรณ์การแพทย์ บริษัทผู้พัฒนาต้องนำกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์เข้ารับการประเมินเพื่อตรวจสอบย้อนกลับกระบวนการพัฒนาซอฟต์แวร์เทียบกับข้อกำหนดมาตรฐาน ISO/IEC 62304 ให้ผ่านก่อนเพื่อควบคุมคุณภาพของซอฟต์แวร์อุปกรณ์การแพทย์ ในงานวิจัยนี้

ตารางที่ 1 โมเดลแสดงข้อกำหนดเพื่อตรวจสอบย้อนกลับกระบวนการพัฒนาซอฟต์แวร์ ส่วนที่ 9 ตามมาตรฐาน ISO/IEC 62304

| Clause | Description                        | Standard Operating Procedure (SOP)  | Plan                             | Record  | Document   |
|--------|------------------------------------|---|----------------------------------|---|--|
| 9.1    | Prepare problem reports            | Software Problem Resolution   | Problem Resolution Plan          |   | Software Problem Resolution report   |
| 9.2    | Investigate the problem            | Software Problem Resolution   |                                  | Problem Resolution Record   | Software Problem Resolution report   |
| 9.3    | Advise relevant parties            | <ul style="list-style-type: none"> <li>Software Problem Resolution</li> <li>Incident Reporting</li> </ul>         | Incident Reporting plan          | Incident Reporting record   | <ul style="list-style-type: none"> <li>Software Problem Resolution report</li> <li>Incident Reporting document.</li> </ul>                                 |
| 9.4    | Use change control process         | Change Management   |                                  | Change request record   | Change management document   |
| 9.5    | Maintain records                   | Software Problem Resolution   |                                  | <ul style="list-style-type: none"> <li>Change request record</li> <li>Change management record</li> </ul>   | Risk Evaluation document   |
| 9.6    | Analyse problem for trends         | Software Problem Resolution   |                                  | Problem record with problem trends detect   |  |
| 9.7    | Verify software problem resolution | <ul style="list-style-type: none"> <li>Software Problem Resolution</li> <li>Change Management (CI 9.4)</li> </ul> | Software Maintenance Plan (CI 6) | <ul style="list-style-type: none"> <li>รายละเอียดปัญหาที่พบ (Problem record)</li> <li>มติการประชุมอนุมัติการแก้ไขปัญหา</li> <li>ผลการทดสอบ</li> </ul> | <ul style="list-style-type: none"> <li>รายงานการแก้ไขปัญหา (Problem report)</li> <li>แบบฟอร์มติดตามสถานะการแก้ไขความคิดปดหรือปัญหาที่พบ</li> </ul>         |
|        |                                    | <p>เกณฑ์การพิจารณากระบวนการมีแนวทาง ขั้นตอนการปฏิบัติได้จริง</p>  |                                  | <p>เกณฑ์การพิจารณา</p> <ul style="list-style-type: none"> <li>ผลการทดสอบ</li> <li>ผลการทดสอบการถอดของการทำงานซอฟต์แวร์ไม่มีอาการถดถอย</li> </ul>      | <p>เกณฑ์การพิจารณา</p> <ul style="list-style-type: none"> <li>ปัญหาถูกปิดจริง</li> <li>มีการดำเนินการแก้ไขปัญหาลงกับซอฟต์แวร์และอุปกรณ์การแพทย์</li> </ul> |

ได้นำเสนอแนวทางการสร้างโมเดลที่ประกอบด้วยกระบวนการขั้นตอนการดำเนินการ (SOP), แผน (PLAN), รายการข้อมูลบันทึกเป็นหลักฐาน (RECORD) และเอกสาร (DOCUMENT) เพื่ออธิบายข้อกำหนดที่ใช้เป็นเกณฑ์การตรวจสอบย้อนกลับของกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ว่าเป็นไปตามมาตรฐาน ISO/IEC 62304 โดยยกตัวอย่างการพัฒนาซอฟต์แวร์ตามกระบวนการแบบจำลองตรวจสอบย้อนกลับเทียบกับข้อกำหนดในส่วนที่ 9 (Problem Resolution)

แนวทางวิจัยในอนาคตแนวทางที่เสนอสามารถนำไปใช้ในการสร้างโมเดลสำหรับการตรวจสอบย้อนกลับของกระบวนการพัฒนาซอฟต์แวร์อุปกรณ์การแพทย์ตามมาตรฐาน ISO/IEC 62304 ในส่วนอื่นๆ และควรมีการทดสอบโมเดลไปใช้ประเมินมาบริษัทผู้ผลิตซอฟต์แวร์อุปกรณ์การแพทย์ในประเทศไทยที่มาใช้บริการจากหน่วยงานงานวิศวกรรมซอฟต์แวร์และทดสอบผลิตภัณฑ์ (SEPT)

ตารางที่ 2 ตัวอย่าง SOP สำหรับกระบวนการ Software Problem Resolution ตามข้อกำหนดส่วนที่ 9.7

| ขั้นตอนที่ | รายละเอียดการทำงาน   |
|------------|--|
| 1          | 1.1 เมื่อพบความคิดปดหรือปัญหาที่พบในระบบซอฟต์แวร์ ให้บันทึกลงในแบบฟอร์มรายงานการพบความคิดปดหรือปัญหา   |
| 2          | 2.1. ตรวจสอบสาเหตุของปัญหา วิเคราะห์ความเสี่ยงที่เกี่ยวข้องกับการใช้งาน อ้างอิงมาตรฐาน ISO 14971<br>2.2. บันทึกผลลงในแบบฟอร์มรายงานการพบความคิดปดหรือปัญหา                               |
| 3          | 3.1. แจ้งความคิดปดหรือปัญหาที่พบให้กับผู้ที่เกี่ยวข้องดำเนินการทางโทรศัพท์ หรือ อีเมล  |
| 4          | 4.1. ดำเนินการ Change request อ้างอิงกระบวนการจัดการคำขอซอฟต์แวร์ (CI. 8 Software Configuration Management Process)  |
| 5          | 5.1. ติดตามสถานการณ์การดำเนินการแก้ไข ให้บันทึกผลลงในแบบฟอร์มติดตามสถานะการแก้ไขความคิดปดหรือปัญหา   |
| 6          | 6.1. ดำเนินการทวนสอบ (Verification) การแก้ไขปัญหาระบบซอฟต์แวร์<br>6.2. บันทึกผลลงในแบบฟอร์มรายงานผลการทดสอบ โดยอ้างอิงกระบวนการแก้ไขปัญหาซอฟต์แวร์ (Software problem resolution process) |
| 7          | 7.1. แจ้งผู้ใช้งานหลังเสร็จสิ้นการแก้ไขทางโทรศัพท์ หรือ อีเมล<br>7.2. กรณีที่ต้องส่งมอบอุปกรณ์หลังการแก้ไข ให้บันทึกผลลงในเอกสารแบบฟอร์มการส่งมอบผลิตภัณฑ์                               |

ตารางที่ 3 เกณฑ์การประเมินส่วนที่ 9.7 มาตรฐาน ISO/IEC 62304

| ข้อ | เกณฑ์การประเมิน  |
|-----|--|
| 9.7 | ผู้ผลิตตรวจสอบกระบวนการแก้ไขปัญหาที่ใช้ว่ามีคุณสมบัติดังต่อไปนี้หรือไม่:   |
|     | ก) มีการจัดทำรายงานแสดงให้เห็นว่าปัญหาได้รับการแก้ไขและปัญหาเบ็ดเสร็จเรียบร้อย โดยมือเอกสารประกอบ อาทิเช่น รายงานการพบความคิดปดหรือปัญหา มติการประชุมในการอนุมัติการแก้ไขปัญหา แบบฟอร์มติดตามสถานการณ์การแก้ไขความคิดปดหรือปัญหาที่พบ รายงานผลการทดสอบ |
|     | ข) มีการบันทึกแสดงหลักฐานให้เห็นแนวโน้มที่จะเกิดขึ้นใหม่ซึ่งประสงคที่เกิดจากปัญหาเป็นไปในทิศทางลดลง  |
|     | ค) คำขอเปลี่ยนแปลงได้รับการดำเนินการกับซอฟต์แวร์และมีการดำเนินการที่เหมาะสมกับอุปกรณ์ทางการแพทย์   |
|     | ง) มีการตรวจสอบซอฟต์แวร์ทุกส่วนว่าไม่เกิดปัญหาการถดถอยของการทำงานซอฟต์แวร์ที่เกิดจากการเปลี่ยนแปลงซอฟต์แวร์ (Regression test)  |

### **เอกสารอ้างอิง**

- [1] F. McCaffery, M. Lepmets, K. Trektere, O. Ozcantop, and M. Pikkarainen, "Agile Medical Device Software Development: Introducing Agile Practices into MDevSPICE," *International Journal on Advances in Life Sciences*, vol. 8, no. 1 & 2, pp. 133-142, 2016 2016.
- [2] M. Mc Hugh, F. McCaffery, and V. Casey, "Barriers to using agile software development practices within the medical device industry," 2012 2012.
- [3] N. M. A. Munassar and A. Govardhan, "A comparison between five models of software engineering," *International Journal of Computer Science Issues (IJCSI)*, vol. 7, no. 5, p. 94, 2010.
- [4] B. P. Douglass and L. Ekas, "Adopting agile method for safety-critical systems development," *IBM Software thought leadership white paper*, no. October, pp. 1-12, 2012.
- [5] ISO. "IEC 62304:2006/Amd 1:2015 Medical device software — Software life cycle processes — Amendment 1." <https://www.iso.org/standard/64686.html> (accessed 2022-08-17 08:59:47).
- [6] "ISO/IEC 15504-5:2012 Information technology — Process assessment — Part 5: An exemplar software life cycle process assessment model." <https://www.iso.org/standard/60555.html> (accessed 2022/08/17/08:25:08).
- [7] D. S. Odaibo, "IEC 62304: Medical Device Software LifeCycle Processes," in *The Blog of RETINA-AI Health, Inc.*, ed, 2021.
- [8] B. Charoendouysil, "มาตรฐานซอฟต์แวร์เครื่องมือแพทย์ (ตอนที่ 3)," in *ห้องปฏิบัติการทดสอบซอฟต์แวร์ SQUAT*, ed, 2021.
- [9] "กองควบคุมเครื่องมือแพทย์, สำนักงานคณะกรรมการอาหารและยา." <https://www.fda.moph.go.th/sites/Medical/Pages/Main.aspx> (accessed 2022-08-23 08:40:38).