บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีต่างๆ ที่นำมาสนับสนุน ประยุกต์ใช้และอ้างอิงในวิทยานิพนธ์ฉบับนี้ รวมถึงงานวิจัยที่เกี่ยวข้องต่างๆ โดยมีรายละเอียดดังต่อไปนี้

2.1. ทฤษฎีที่เกี่ยวข้อง

งานวิจัยนี้ได้นำเอาการค้นคืนสารสนเทศมาช่วยในการค้นคืนแบบรูปความต้องการด้านความมั่นคง ซึ่งได้มีการนำทฤษฎีที่เกี่ยวข้องได้แก่ การค้นคืนสารสนเทศ วิศวกรรมความต้องการซอฟต์แวร์ ความต้องการด้านความมั่นคง แบบรูปและแบบรูปสำหรับซอฟต์แวร์ และสุดท้ายคือ แบบรูปความมั่นคง โดยมีส่วนของรายละเอียดดังต่อไปนี้

2.1.1. การจัดเก็บและค้นคืนสารสนเทศ (Information Storage and Retrieval)

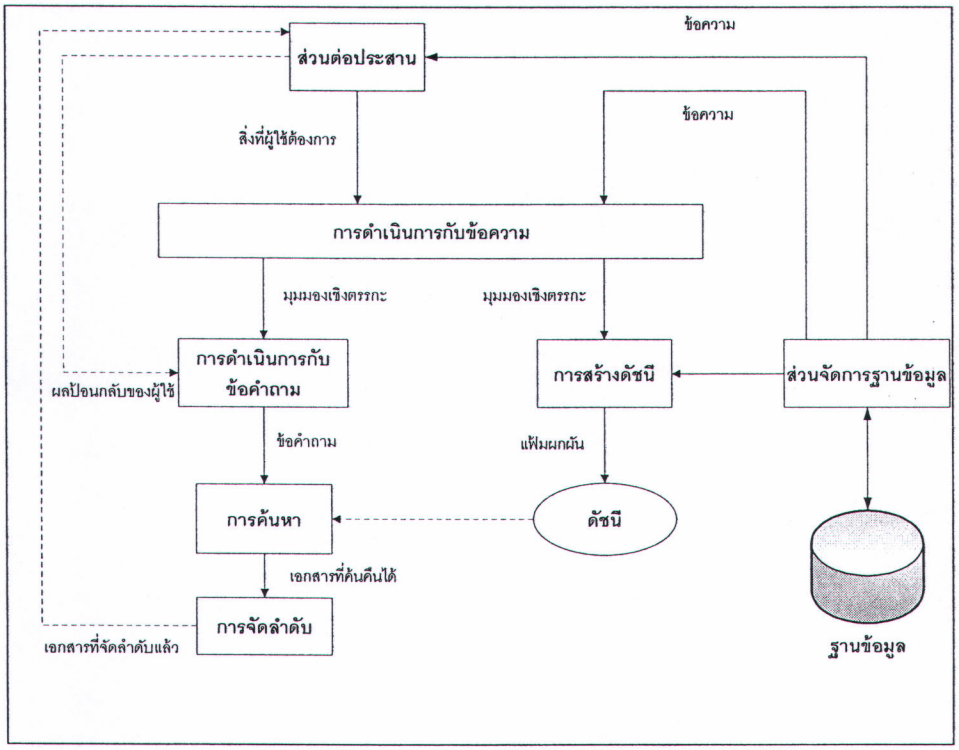
การค้นคืนสารสนเทศ [7, 8] คือกระบวนการในการค้นคืนสารสนเทศที่ได้มีการจัดเก็บอยู่ในแหล่งเก็บ เพื่อให้ได้สารสนเทศที่ตรงกับความต้องการของผู้ใช้มากที่สุดโดยในการค้นคืนสารสนเทศนั้นมีแบบจำลองอยู่หลักๆ ดังแบบจำลองดังต่อไปนี้

(1) แบบจำลองตรรกะ (Boolean Model) เป็นตัวแบบที่จะทำการเทียบดูคำที่ปรากฏในเอกสารเพียงว่ามีหรือไม่มีเท่านั้นโดยมีการใช้ตัวดำเนินการ และ (and) หรือ (or) ไม่ (not) ในการเทียบคำในเอกสารเช่น หากใส่คำถามเข้าไปทั้งสิ้น 2 คำ หากใช้ตัวดำเนินการและหากไม่พบคำใดคำหนึ่งก็ถือว่าเอกสารนั้นไม่ตรงความต้องการในทันที

(2) แบบจำลองปริภูมิเวกเตอร์ (Vector Space Model) เป็นตัวแบบที่พัฒนาต่อมาจากตัวแบบตรรกะ โดยมีการจัดทำการใช้ความถี่ที่ปรากฏในเอกสาร และค่าน้ำหนักในการคำนวณจึงได้ค่าที่หลากหลายกว่าการทำการเทียบดูเพียงแค่ว่ามีหรือไม่มีดังตัวแบบตรรกะ ซึ่งจะผลที่ได้ทำให้สามารถจัดลำดับของสิ่งที่ทำการค้นคืนมาได้ โดยในงานวิจัยนี้ได้ใช้แบบจำลองปริภูมิเวกเตอร์นี้ในการทำวิจัย

(3) แบบจำลองความน่าจะเป็น (Probabilistic Model) เป็นตัวแบบที่ใช้หลักของความน่าจะเป็นในคำนวณเพื่อทำการค้นคืนค่าโดยให้แนวทางว่า หากค่าความน่าจะเป็นของความเกี่ยวเนื่องกันของคำถามและเอกสารมีค่าสูงแล้ว เอกสารนั้นก็ควรที่จะมีความใกล้เคียงกับคำถามที่ผู้ใช้ให้เข้ามามากเช่นกัน

ระบบการค้นคืนสารสนเทศโดยทั่วไปมีกระบวนการดังต่อไปนี้ เริ่มโดยผู้ใช้จะทำการป้อนคำถาม (Query) เข้าสู่ระบบแล้วระบบจะทำการสร้างดัชนีของคำถาม และนำดัชนีที่ได้ไปทำการคำนวณค่าความคล้ายกัน (Similarity) ระหว่างคำถามที่ได้ป้อนไปกับเอกสารที่ได้ถูกจัดเก็บอยู่ในระบบ โดยจะนำเสนอรายการของผลลัพธ์โดยทำการเรียงลำดับซึ่งลำดับของผลลัพธ์จะขึ้นอยู่กับความคล้ายคลึงกันของระหว่างคำถามกับตัวเอกสารนั้นๆ และหากรายการผลลัพธ์ที่ได้มาไม่ตรงกับความต้องการของผู้ใช้ ผู้ใช้ก็สามารถทำการผลป้อนกลับเพื่อทำการปรับปรุงคำถามให้ได้ผลที่ดียิ่งขึ้น ซึ่งแสดงกระบวนการค้นคืนสารสนเทศดังรูปที่ 2.1



รูปที่ 2.1 กระบวนการจัดเก็บและค้นคืนสารสนเทศ [8]

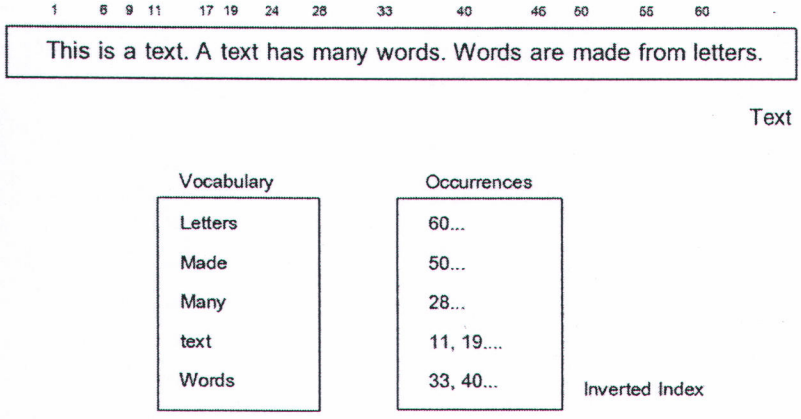
โดยมีรายละเอียดของแต่ละกระบวนการดังต่อไปนี้

- ส่วนต่อประสาน (User Interface) คือส่วนที่ทำหน้าที่ในการเป็นส่วนติดต่อระหว่าง ระบบกับผู้ใช้ โดยมีการรับสิ่งที่ผู้ใช้ต้องการสู่ระบบ และแสดงผลลัพธ์จากการทำงานของระบบให้แก่ผู้ใช้

- การดำเนินการกับข้อความ (Text Operations) คือส่วนในการแปลงเอกสารเดิมเพื่อสร้างเป็นดัชนี ซึ่งประกอบไปด้วยขั้นตอนต่างๆ เพื่อช่วยทำให้เอกสารแปลงรูปเป็นดัชนีของเอกสาร โดยจะกล่าวถึงขั้นตอนในหัวข้อ 2.1.2

- การดำเนินการกับข้อคำถาม (Query Operations) คือส่วนในการแปลงข้อคำถามที่ได้จากผู้ใช้ โดยจะทำให้อยู่ในลักษณะของดัชนี เช่นเดียวกับในส่วนการดำเนินการกับข้อความ

- การสร้างดัชนี (Indexing) คือการกระบวนการในการจัดทำดัชนีที่ได้จากการดำเนินการกับข้อความมาจัดเก็บให้อยู่ในรูปโครงสร้างซึ่งวิธีที่ได้รับความนิยมมากที่สุดได้แก่ แฟ้มผกผัน (Inverted file) โดยองค์ประกอบของโครงสร้างแฟ้มผกผันมีทั้งสิ้น 2 องค์กอบได้แก่ กลุ่มคำศัพท์ (Vocabulary) กับ เหตุการณ์ (Occurrences) ตัวอย่างการจัดทำแฟ้มผกผันแสดงดังรูปที่ 2.2



รูปที่ 2.2 ตัวอย่างการจัดทำข้อความเป็นแฟ้มผกผัน [8]

- การค้นหา (Searching) คือการทำการหาค่าความคล้ายคลึงระหว่างคำถามที่รับเข้ามากับดัชนีที่จัดเก็บอยู่ในฐานข้อมูล

- การจัดลำดับ (Ranking) คือการทำการจัดลำดับความคล้ายคลึงที่ได้จากการคำนวณความคล้ายคลึงเพื่อนำเสนอเอกสารที่มีความคล้ายคลึงมากไปสู่เอกสารที่มีความคล้ายคลึงน้อย

- ส่วนจัดการฐานข้อมูล (DB Manager Module) คือส่วนในการติดต่อฐานข้อมูลในการจัดเก็บข้อความ ดัชนี และการดึงข้อความขึ้นมาแสดงผล

จากรูปที่ 2.1 ได้แสดงภาพรวมของการทำงานของกระบวนการจัดเก็บและค้นคืน ซึ่งในส่วนการทำงานภายในจะกล่าวถึงในส่วนถัดไป

2.1.1.1. การจัดเก็บเอกสาร

ในการจัดเก็บเอกสารในระบบสารสนเทศนั้นมีการจัดเก็บได้ 3 แบบ

(1) การจัดเก็บแบบเชิงเส้น (Linear Lists) เป็นการจัดเก็บแบบต่อท้ายไปเรื่อยๆ

(2) การจัดเก็บแบบเรียงลำดับ (Order sequential Files) เป็นการจัดเก็บแบบการเรียงลำดับที่ได้กำหนดในการจัดเก็บ เช่น เรียงตามชื่อ

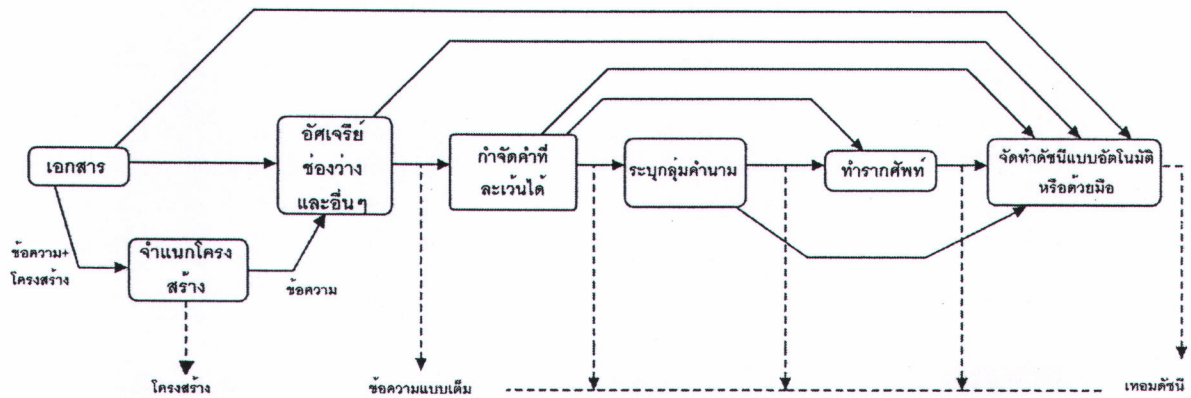
(3) การจัดเก็บแบบดัชนี (Index) เป็นการจัดเก็บเอกสารโดยจัดเก็บตามคำสำคัญ โดยสามารถที่จะแบ่งย่อยได้เป็นแบบแฟ้มตรง (Directed File) และ แบบผกผัน (Inverted File)

โดยในงานวิจัยนี้ได้ใช้วิธีการจัดเก็บแบบดัชนีในการช่วยจัดเก็บเอกสารโดยการทำการจัดทำจัดเก็บเอกสารตามคำสำคัญเพื่อนำไปใช้ในการค้นคืนต่อไป

2.1.1.2. การสร้างดัชนีในการจัดเก็บเอกสารและการคำนวณค่าน้ำหนักของดัชนี

ดัชนีของเอกสารคือคำที่สามารถระบุถึงเอกสารที่เราต้องการที่จะทำการค้นคืนหากมีคำถามร้องขอเข้าสู่ระบบซึ่งในหนึ่งเอกสารจะมีหลายดัชนีในการระบุถึง

โดยในการสร้างดัชนีในการจัดเก็บเอกสารเป็นกระบวนการสำคัญที่มีผลต่อการค้นคืนในภายภาคหน้า เพราะหากทำการสร้างดัชนีที่ไม่ตรงกับเอกสาร หรือ หากให้น้ำหนักในคำที่ไม่สำคัญมากกว่าคำที่มีความสำคัญหรือ คำหลัก (Keyword) จะทำให้ระบบการค้นคืนผิดพลาดหรือ ระบบอาจใช้การไม่ได้เลย โดยมีกระบวนการการจัดทำดัชนีดังรูปที่ 2.3



รูปที่ 2.3 กระบวนการในการสร้างดัชนีเชิงตรรกะ [8]

จากรูปที่ 2.3 เริ่มจากการนำเอกสารเข้ามา มีกระบวนการต่างๆ มากมายโดยอธิบายเรียงลำดับได้ดังนี้

- ทำการจำแนกโครงสร้างของเอกสาร (Structure recognition) เช่นหนังสือ มีโครงสร้างดังนี้ บท หัวข้อ ข้อย่อย และอื่นๆ เป็นต้นทำให้ได้ตัวโครงสร้าง (Structure) ของเอกสารและตัวข้อความ (Text) ของเอกสาร

- หลังจากการทำการจำแนกโครงสร้างเสร็จสิ้นจึงทำการกำจัดเครื่องหมายต่างๆ ดังเช่น '.', ',', '?', '!' ออกจากตัวข้อความ

- ทำการกำจัดคำที่สามารถละเว้นได้ (stopwords) โดยคำกลุ่มนี้มักเป็นคำที่สามารถพบได้บ่อยๆ เช่น 'is', 'am, 'are', 'and' เป็นต้น

- ทำการระบุกลุ่มคำนาม (noun groups) ได้แก่ คำนามคำเดียว (single noun) สองคำใกล้กัน (two adjacent nouns) หรือ สามคำใกล้กันเป็นต้น (three adjacent nouns)

- ทำการหารากศัพท์ (stemming) เป็นการทำหารากของคำศัพท์นั้นๆ เช่น 'shopping' จะถูกแปลงเป็น 'shop' เพื่อทำคำที่ได้ไปทำดัชนีต่อไป โดยในงานวิจัยนี้ได้ใช้การทำรากศัพท์โดยขั้นตอนวิธีการของพอร์ตเตอร์ (Porter's Algorithm)

ซึ่งเมื่อได้ดัชนีของเอกสารนั้นๆ แล้ว การใช้แบบจำลองปริภูมิเวกเตอร์นั้น มีการให้ค่าน้ำหนักของคำที่ปรากฏในเอกสารโดยการคำนวณน้ำหนักของดัชนีต่อเอกสารใช้สมการ [8] ดังต่อไปนี้

$$w_{i,j} = tf_{i,j} \times idf_i \tag{1}$$

เมื่อ $w_{i,j}$ แทน น้ำหนักของนิพจน์ i นั้นในเอกสาร j

$tf_{i,j}$ แทน ความถี่ของนิพจน์ i ที่ปรากฏในเอกสาร j

idf_i แทน ค่าผกผันของความถี่ของเอกสารที่มีนิพจน์ i ปรากฏอยู่

สมการในการคำนวณค่าความถี่ที่ปรากฏในเอกสาร (Term frequency) แสดงดังสมการ [8] ดังต่อไปนี้

$$tf_{i,j} = \frac{freq_{i,j}}{(\max_l freq_{l,j})} \tag{2}$$

เมื่อ $tf_{i,j}$ แทน ค่าความถี่ที่ได้จากการคำนวณของนิพจน์ i ที่ปรากฏในเอกสารที่ j

$freq_{i,j}$ แทน ความถี่จริงๆ ของนิพจน์ i ที่ปรากฏในเอกสาร j

$\max_l freq_{l,j}$ แทน ความถี่สูงสุดของนิพจน์ที่ปรากฏในเอกสาร j

โดยสมการในการคำนวณค่าความถี่ที่ปรากฏในเอกสารโดยวิทยานิพนธ์ฉบับนี้อ้างอิงตามหนังสือการค้นคืนข้อมูลแบบทันสมัย (Modern Information Retrieval) [8] ซึ่งในหนังสือเล่มอื่นอาจมีสูตรการคำนวณความถี่ของคำที่ปรากฏในเอกสารแตกต่างไปจากนี้

สมการในการคำนวณค่าผกผันของความถี่ของเอกสาร (Inverse document frequency: idf) แสดงดังสมการ [8] ดังต่อไปนี้

$$idf_i = \log \frac{N}{n_i} \tag{3}$$

เมื่อ idf_i แทน ค่าผกผันของความถี่ของเอกสารที่มีนิพจน์ i ปรากฏ

N แทน จำนวนเอกสารทั้งหมดในระบบ

n_i แทน จำนวนเอกสารทั้งหมดที่มีนิพจน์ i ปรากฏ

2.1.1.3. การคำนวณความคล้ายคลึงกัน

เป็นกระบวนการในการค้นคืนเอกสารโดยทำการคำนวณค่าความคล้ายคลึงของคำถามของผู้ใช้กับตัวเอกสารว่ามีความคล้ายคลึงกันเท่าใด โดยในที่นี้ใช้หลักการในการคำนวณความคล้ายคลึงของแบบจำลองปริภูมิเวกเตอร์ ได้แก่ การวัดความคล้ายคลึงแบบโคไซน์ (Cosine Similarity) มาคำนวณค่าความคล้ายคลึงระหว่างตัวคำถาม (q) กับ เอกสารที่ j โดยใช้สมการ [7, 8]ดังต่อไปนี้

$$sim(d_j, q) = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \tag{4}$$

เมื่อ $sim(d_j, q)$ แทน ค่าความคล้ายคลึงระหว่างเอกสารที่ j กับคำถาม

$w_{i,j}$ แทน ค่าน้ำหนักของนิพจน์ i ในเอกสารที่ j

$w_{i,q}$ แทน ค่าน้ำหนักของนิพจน์ i ในคำถามที่ป้อนเข้าไป

t แทน จำนวนนิพจน์ที่ใช้ในการคำนวณ

2.1.1.4. การประเมินประสิทธิผลของการค้นคืน

การประเมินประสิทธิผลของกระบวนทำการค้นคืน สามารถทำการเปรียบเทียบโดยใช้ ค่าความแม่นยำ (Precision) กับ ค่าระลึก (recall) โดยสมการค่าความแม่นยำแสดงในสมการที่ 5 [7, 8] และ สมการค่าระลึกในสมการที่ 6 [7, 8]

$$\text{ค่าความแม่นยำ} = \frac{|Ra|}{|A|} \quad (5)$$

$$\text{ค่าระลึก} = \frac{|Ra|}{|R|} \quad (6)$$

เมื่อ $|Ra|$ แทน จำนวนเอกสารที่ตรงตามความต้องการของผู้ใช้

$|A|$ แทน จำนวนเอกสารทั้งหมดที่ได้ถูกทำการค้นคืนขึ้นมา

$|R|$ แทน จำนวนเอกสารทั้งหมดที่ผู้ชำนาญเป็นผู้กำหนดให้กับฐานข้อมูลที่ตรงกับข้อคำถาม

โดยนอกจากมาตรวัดทั้ง 2 แบบที่ได้กล่าวไปแล้ว ยังมีมาตรวัดอีกอันหนึ่ง ที่นำทั้งสองมาตรวัดที่ได้กล่าวมาแล้วมาสร้างเป็นมาตรวัดที่ได้เป็นตัวเลขเดียวทำให้ง่ายต่อการพิจารณา ซึ่งก็คือ ค่าเฉลี่ยฮาร์โมนิค (Harmonic mean) ซึ่งได้ผลลัพธ์จากการค้นคืนอยู่ในช่วง 0 ถึง 1 โดยยิ่งเข้าใกล้ 1 เท่าไรก็จะยิ่งมีความคล้ายคลึงมากขึ้น โดยสมการแสดงดังสมการที่ 7 [7, 8] ดังต่อไปนี้

$$\text{ค่าเฉลี่ยฮาร์โมนิค } F(j) = \frac{2}{\frac{1}{r(j)}+\frac{1}{P(j)}} \quad (7)$$

เมื่อ $F(j)$ แทน ค่าเฉลี่ยฮาร์โมนิค

$r(j)$ แทน ค่าระลึกของเอกสารลำดับที่ j

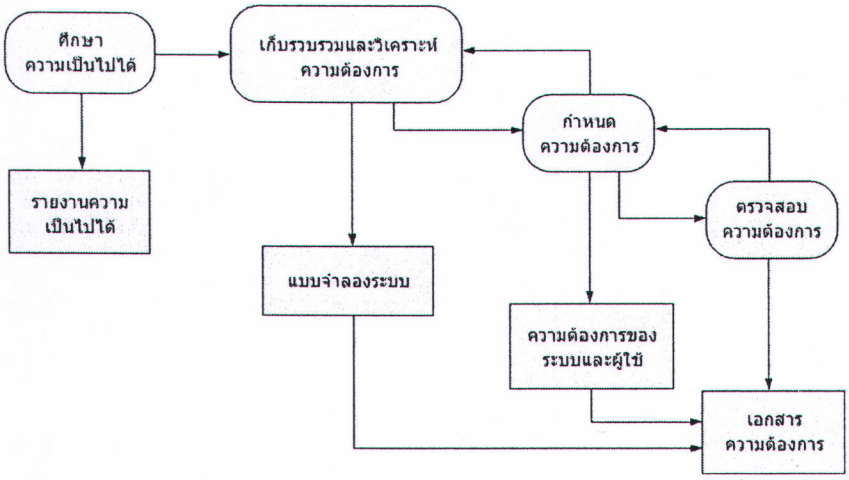
$P(j)$ แทน ค่าความแม่นยำของเอกสารลำดับที่ j

j แทน ลำดับของเอกสารที่ j

2.1.2. **วิศวกรรมความต้องการซอฟต์แวร์** (Software Requirements Engineering)

วิศวกรรมความต้องการซอฟต์แวร์เป็นส่วนหนึ่งของวิศวกรรมซอฟต์แวร์ ถูกกำหนดขึ้นในช่วงการเริ่มต้นของกระบวนการพัฒนาซอฟต์แวร์ โดยมีจุดหมายในการให้ได้มาซึ่ง

ความต้องการด้านซอฟต์แวร์ที่ถูกต้อง และชัดเจนเพื่อนำไปใช้ในการกำหนดระบบที่จะทำการพัฒนา โดยมีกระบวนการสำคัญดังรูปที่ 2.4 มีดังนี้



รูปที่ 2.4 กระบวนการในการทำวิศวกรรมความต้องการ [9]

จากรูปที่ 2.4 กระบวนการในการทำวิศวกรรมความต้องการมีคำบรรยายกระบวนการดังนี้ [9, 10]

1) ศึกษาความเป็นไปได้ (Feasibility studies) เป็นการศึกษาความเป็นไปได้ของระบบใหม่ที่จะทำการพัฒนา ซึ่งจะทำการนำเอาความต้องการทางธุรกิจเบื้องต้น รายละเอียดโครงร่างตัวระบบ และจะนำเอาระบบไปสนับสนุนธุรกิจอย่างไรเข้ามาเป็นส่วนของสิ่งนำเข้า ซึ่งผลลัพธ์จากการศึกษาความเป็นไปได้คือ รายงานความเป็นไปได้ เพื่อนำไปเสนอเพื่อให้เข้าสู่กระบวนการทางวิศวกรรมความต้องการต่อไป

2) การเก็บรวบรวมความต้องการ (Requirements Elicitation) เป็นกระบวนการที่จะทำการรวบรวมความต้องการโดยใช้เทคนิคต่างๆ ในการรวบรวมความต้องการเช่นการสัมภาษณ์ การรวมรวมเอกสาร หรือเทคนิคอื่นๆ ซึ่งต้องเข้าไปเกี่ยวข้องกับผู้ใช้ระบบโดยตรง โดยมีจุดประสงค์ในการเก็บข้อมูลที่จะนำมาใช้ในการกำหนดตัวระบบให้ได้ความครบถ้วน ความถูกต้องของตัวข้อกำหนดความต้องการ

3) การวิเคราะห์ความต้องการ (Requirements Analysis) เป็นกระบวนการวิเคราะห์ความต้องการว่าระบบที่ต้องการพัฒนานั้นมีความเกี่ยวเนื่องกับผู้เกี่ยวข้องใครบ้าง ระบบใดบ้าง หรือ ถูกใช้เมื่อไร ถูกใช้โดยใคร โดยอาจจะต้องมีการกำหนดสิทธิ์ต่างๆ ในการใช้งานระบบ

4) การระบุข้อกำหนดความต้องการ (Requirements Specification) เป็นกระบวนการใน การจัดทำข้อกำหนดตัวระบบ ให้ได้รายระเอียดของสิ่งที่เราต้องทำการพัฒนาให้อยู่ในรูปแบบที่เป็นตามรูปแบบที่สอดคล้องกับข้อกำหนดต่างๆ ซึ่งเป็นผลลัพธ์ที่สำคัญที่ได้จากกระบวนการวิศวกรรมความต้องการซอฟต์แวร์

5) การตรวจสอบความต้องการ (Requirements Validation) เป็นการกระบวนการในการตรวจสอบความถูกต้องของความต้องการที่ได้จัดทำขึ้นมา เช่น ความต้องการที่เก็บมาได้นั้นมีความสอดคล้องกันหรือไม่ มีการจัดระดับความสำคัญของความต้องการอย่างไร

ซึ่งนอกจากกิจกรรมข้างต้นยังมีอีกหนึ่งกิจกรรมที่สำคัญต่อกระบวนการวิศวกรรมความต้องการได้แก่ การจัดการความต้องการ (Requirements Management) ซึ่งเป็นส่วนในการบริหารกระบวนการวิศวกรรมความต้องการซอฟต์แวร์ จัดการดูแลกิจกรรมต่างๆ การเก็บรวบรวมความต้องการ การวิเคราะห์ความต้องการ การระบุข้อกำหนดความต้องการ การตรวจสอบความต้องการ รวมถึงการติดตามเปลี่ยนแปลงความต้องการ (Requirements Change) และการตามรอยความต้องการ (Requirements Tracing) เป็นต้น

ความต้องการนั้นสามารถแบ่งออกได้เป็น 2 ประเภทหลักๆ ได้แก่

(1) ความต้องการเชิงหน้าที่ เป็นความต้องการที่มุ่งไปที่กระบวนการต่างๆ ของระบบที่ต้องกระทำให้ได้หรือเรียกได้ว่าเป็นหน้าที่หลักของระบบ

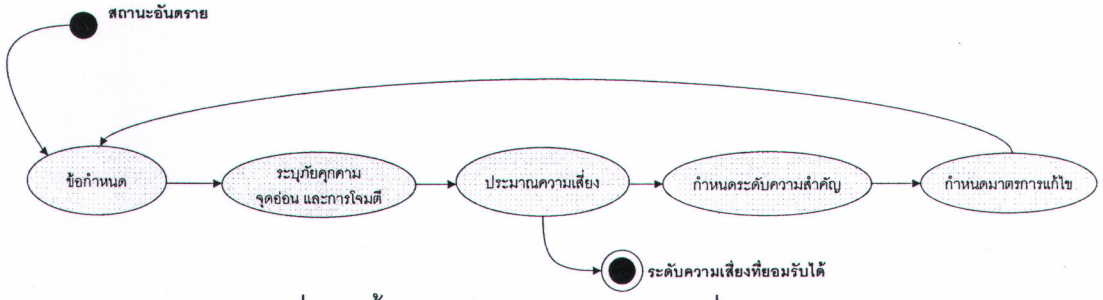
(2) ความต้องการที่ไม่ใช่หน้าที่ เป็นความต้องการที่อาจไม่ได้ระบุให้ชัดเจน เช่น ประสิทธิภาพ ความมั่นคง และความปลอดภัย เป็นต้น ซึ่งมักใช้สำหรับตัดสินคุณภาพของระบบ

โดยกระบวนการวิศวกรรมความต้องการซอฟต์แวร์นั้นถือเป็นกระบวนการที่สำคัญมากในกระบวนการพัฒนาซอฟต์แวร์เนื่องมาจากหากมีการบริหารจัดการที่ไม่ดีเพียงพอจะทำให้เกิดความผิดพลาดตามมาในกระบวนการพัฒนาซอฟต์แวร์ที่จะตามมา ได้แก่ การพัฒนาซอฟต์แวร์ การทดสอบซอฟต์แวร์ การนำไปใช้และการดูแลรักษา ซึ่งหากเกิดความผิดพลาดหรือไม่สมบูรณ์ที่กระบวนการวิศวกรรมความต้องการจะทำให้เสียค่าใช้จ่ายในการปรับปรุงแก้ไขระบบ ทำให้งานที่พัฒนาไม่เสร็จตรงตามเวลาที่ได้กำหนดไว้ รวมถึงการเกิดข้อบกพร่อง หรือข้อผิดพลาดที่เกิดจากการรวบรวมและกำหนดความต้องการบกพร่อง ซึ่งวัตถุประสงค์ของวิศวกรรมความต้องการซอฟต์แวร์นั้น เพื่อให้ได้ ผลลัพธ์ 3 สิ่ง คือ การยอมรับในความต้องการ

(Agreed Requirements) ข้อกำหนดของระบบ (System Specification) และแบบจำลองระบบ (System Model)

2.1.3. ความต้องการด้านความมั่นคง (Security Requirements)

ความต้องการความมั่นคง [11] ถือเป็นความต้องการที่ไม่ใช่หน้าที่ ซึ่งเป็นความต้องการเชิงคุณภาพในด้านความมั่นคง โดยมุ่งเน้นในการป้องกันภัยอันตรายที่จะเกิดขึ้นกับระบบหรือการใช้งานผิดไปจากที่ควรจะเป็น ดังนั้นจึงควรมีการจัดทำความต้องการความมั่นคงให้ชัดเจน เพื่อจัดการสถานะอันตรายของสิ่งที่ต้องการจะปกป้องให้อยู่ในระดับความเสี่ยงที่ยอมรับได้ โดยแสดงกระบวนการจัดการความเสี่ยงด้วยขั้นตอนวิธีการทางวิศวกรรมความมั่นคงแสดงดังรูปที่ 2.5



รูปที่ 2.5 ขั้นตอนวิธีทางวิศวกรรมความมั่นคง [1]

โดยขั้นตอนวิธีทางวิศวกรรมความมั่นคงดังที่ได้แสดงในรูปที่ 2.5 มีรายละเอียดดังต่อไปนี้ [1]

1) ข้อกำหนด (Specification) เป็นการกำหนดถึงส่วนประกอบ (Components) และ ส่วนต่อประสาน (Interfaces) ของระบบทั้งหมดให้ครบถ้วนสมบูรณ์ เพราะหากถ้าข้อกำหนดของสถาปัตยกรรมของระบบไม่ครอบคลุม ในทุกๆ ส่วนประกอบ อาจเกิดภัยคุกคาม จุดอ่อน และการโจมตี ในส่วนที่ยังไม่ได้ทำการระบุในข้อกำหนดไว้

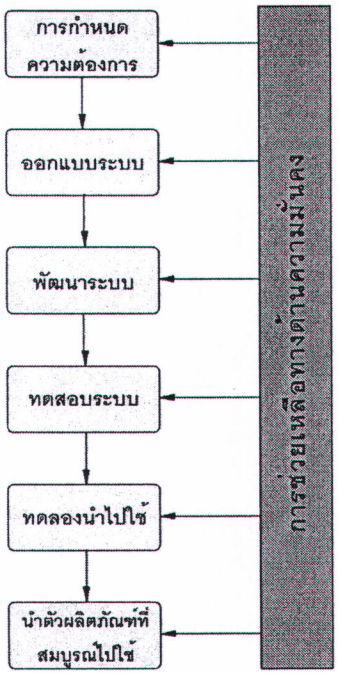
2) ระบุภัยคุกคาม จุดอ่อน และการโจมตี (Identification of Threats, Vulnerabilities and Attacks) เป็นการระบุถึงภัยคุกคามและจุดอ่อนพื้นฐานของในแต่ละส่วนประกอบและส่วนต่อประสานที่ระบุไว้แล้วในข้อกำหนด ซึ่งช่วยในการกำหนดการโจมตีที่คาดว่าจะเกิดขึ้นกับภัยคุกคามและจุดอ่อนที่ระบุได้

3) ประมาณความเสี่ยง (Risk Estimation) เป็นการประมาณถึงความเสี่ยงที่คาดว่าจะโดนโจมตีจะถูกกำหนดสำหรับทุกๆ ส่วนประกอบ และ ส่วนต่อประสาน โดยจะต้องพิจารณาจากผลของความสัมพันธ์ระหว่าง ข้อกำหนดภัยคุกคาม จุดอ่อน และการโจมตี

4) กำหนดระดับความสำคัญ (Prioritization) เป็นการกำหนดระดับให้กับความเสี่ยงที่ได้รับการประเมิน โดยหากพบจุดอ่อนที่เป็นอันตรายต่อส่วนประกอบ หรือส่วนต่อประสานในระดับสูง จะถือว่ามีความเสี่ยงสูง จำเป็นต้องได้รับการดูแลในระดับต้นๆ ซึ่งการเรียงลำดับอย่างมีประสิทธิภาพในขั้นตอนนี้ถือเป็นขั้นตอนที่สำคัญมาก เนื่องจากจะส่งผลต่อมาตรการตอบโต้ในขั้นตอนต่อไป

5) มาตราการตอบโต้ (Countermeasures) เป็นเลือกวิธีการที่เหมาะสมในการกำจัดภัยคุกคาม จุดอ่อน และการโจมตี หรือ ทำให้มีผลกระทบให้น้อยที่สุด ซึ่งก็ขึ้นกับความต้องการของเจ้าของระบบที่จะพิจารณาเลือกใช้เนื่องจากปัจจัยต่าง ได้แก่ ค่าใช้จ่าย ประโยชน์ที่ได้รับ ประสิทธิภาพ หรืออื่นๆ

โดยปัญหาใหญ่ที่สุดของความมั่นคงคือการที่องค์กรส่วนมากเวลาทำโครงการใหม่ขึ้นมามักจะทำการตรวจสอบความมั่นคงภายหลังวัฏจักรการพัฒนา [12] ทำให้ให้เกิดความเสี่ยงและทำให้สิ้นเปลืองงบประมาณในการแก้ไขหรือป้องกันในภายหลัง ซึ่งในความเป็นจริงความมั่นคงนั้นได้ถูกนำไปใช้ตลอดขั้นตอนของการพัฒนาซอฟต์แวร์ดังรูปที่ 2.6



รูปที่ 2.6 ความเกี่ยวข้องระหว่างความมั่นคงกับวิธีการออกแบบที่เหมาะสม [12]

โดยในส่วนการกำหนดความต้องการนั้นเป็นช่วงที่มีความสำคัญมากต่อการพัฒนาซอฟต์แวร์ เนื่องมาจากถ้าความต้องการที่ถูกนำมาเป็นส่วนที่กำหนดรายละเอียดของโครงการ

ไม่ได้มีการระบุรายละเอียดอย่างเหมาะสมแล้ว ระบบที่พัฒนาขึ้นมาจากความต้องการนั้นก็จะไม่สามารถทำงานได้ตามความคาดหวังของผู้ใช้ ซึ่งปัญหาด้านความมั่นคงหากไม่สามารถระบุได้อย่างเหมาะสมในระยะเริ่มต้นของโครงการแล้ว จะเพิ่มความเสี่ยงและค่าใช้จ่ายให้กับโครงการเป็นอย่างมาก โดยเฉพาะอย่างยิ่งหากพบปัญหาด้านความมั่นคง หลังจากได้นำซอฟต์แวร์ไปใช้งานแล้วและจำเป็นที่ต้องมีการปรับปรุงเพื่อแก้ไขปัญหานั้น

2.1.4. **แบบรูปและแบบรูปซอฟต์แวร์** (Patterns and Software Patterns)

แบบรูปคือ แบบแผนหรือแนวทางที่นำมาเพื่อสร้างสิ่งสิ่งหนึ่งขึ้นมา และมักเป็นเหตุการณ์ที่ซ้ำเดิมอีก โดยแบบรูปนั้นจะระบุถึงปัญหาต่างๆ รวมถึงผลเฉลยเอาไว้ด้วยกัน ถูกสร้างมาโดยนักสถาปนิกที่ชื่อว่า Christopher Alexander โดยเริ่มแรกนำมาใช้สำหรับการวางแผนและการก่อสร้างด้านสถาปัตยกรรม ต่อมาได้ถูกนำไปใช้ในหลายๆ โดเมน (domain) รวมถึงในด้านการพัฒนาซอฟต์แวร์ โดยเริ่มต้นที่นาย Ward Cunningham และ Kent Beck ได้นำแบบรูปมาใช้ในการออกแบบส่วนประสานผู้ใช้ในการพัฒนาซอฟต์แวร์ และถูกนำมาใช้งานกันอย่างแพร่หลายเช่น แบบรูปการวิเคราะห์ระบบ แบบรูปการออกแบบ แบบรูปสถาปัตยกรรม เป็นต้น

จากตัวอย่างที่ได้กล่าวขั้นต้นได้ถูกนำมาจัดลำดับของแบบรูปออกตามกิจกรรมของการพัฒนาซอฟต์แวร์ โดยแบ่งออกโดยการนำไปใช้ดังต่อไปนี้ [1]

1) แบบรูปสถาปัตยกรรม (Architectural Patterns) ถูกจัดเป็นระดับสูงของกระบวนการพัฒนาซอฟต์แวร์โดยแสดงถึงความสัมพันธ์ระหว่างองค์กร

2) แบบรูปการออกแบบ (Design Patterns) เป็นระดับกลาง โดยจะใช้กับพวกระบบย่อยหรือพวกส่วนของโปรแกรม (component) โดยใช้ในการแสดงความสัมพันธ์ระหว่างพวกที่ได้กล่าวไป

3) สำนวน (Idioms) เป็นระดับล่างที่สุด โดยจะให้ในการเรียบเรียงแนวทางการแก้ปัญหาให้ได้ตรงตามที่ได้ออกแบบไว้

โดยทั่วไปแบบรูปจะประกอบไปด้วยองค์ประกอบหลักทั้งสิ้น 3 ส่วน ได้แก่ บริบท ปัญหา และ ผลเฉลย แต่ในการนำไปใช้งานนั้นจำเป็นต้องมีการระบุองค์ประกอบอื่นๆ เพิ่มเติมเพื่อความสมบูรณ์และง่ายต่อการนำไปใช้ เช่น ชื่อแบบรูป (Pattern Name) แบบรูปที่เกี่ยวข้อง (Related Patterns) ตัวอย่าง (Example) เป็นต้น ซึ่งแบบรูปที่เป็นที่รู้จักเป็นอย่างดีได้แก่ แบบ

รูปการออกแบบของ E. Gamma และคณะ [13] ที่สร้างแบบรูปการออกแบบสำหรับการพัฒนาซอฟต์แวร์เอาไว้

การนำแบบรูปไปใช้งานนั้นบังเกิดผลดีในหลายๆ ด้านไปแก่ การนำสิ่งที่คนยอมรับและเคยมีอยู่แล้วมาใช้ใหม่นั้นทำให้เกิดการเสี่ยงน้อยกว่าการคิดใหม่ทั้งหมด รวมถึงการทำซ้ำในวิธีการเดิมๆ จะช่วยให้เกิดความเชี่ยวชาญในด้านนั้นๆ อีกด้วย รวมถึงทำให้ผู้ที่มีประสบการณ์ที่แตกต่างกันยังสามารถทำผลลัพธ์ที่ใกล้เคียงกัน ซึ่งช่วยในการรักษาระดับมาตรฐานการพัฒนาซอฟต์แวร์ โดยการลดช่องว่างระหว่างค่าประสบการณ์ของแต่ละบุคคล

2.1.5. **แบบรูปความมั่นคง** (Security Patterns)

แบบรูปความมั่นคง คือแบบแผนหรือแนวทางที่ใช้ในการแก้ปัญหาที่มักเกิดขึ้นเสมอในการออกแบบพัฒนาระบบ โดยมุ่งเน้นไปในปัญหาด้านความมั่นคงซึ่งปรากฏอยู่บ่อยครั้ง โดยแต่ละแบบรูปความมั่นคงนั้นมุ่งที่จะปรับปรุงซอฟต์แวร์ให้ได้รับคุณสมบัติทางด้านความมั่นคงในบางประการ ได้แก่ ความเป็นความลับ บูรณภาพ ความรับผิดชอบ และสภาพพร้อมใช้งาน รวมทั้งสิ้น 4 คุณสมบัติโดยมีรายละเอียดดังต่อไปนี้

1) ความเป็นความลับ (Confidentiality) เป็นคุณสมบัติที่เกี่ยวกับการเปิดเผยความลับเพียงบุคคลหรือหน่วยงานที่ได้รับอนุญาติโดยบริษัท

2) บูรณภาพ (Integrity) เป็นคุณสมบัติที่เกี่ยวกับการที่สินทรัพย์ขององค์กรจะไม่โดนเปลี่ยนแปลงแก้ไขไปจากที่องค์กรต้องการ

3) ความรับผิดชอบ (Accountability) เป็นคุณสมบัติที่เกี่ยวกับการที่สามารถระบุถึงผู้ที่กระทำต่อสินทรัพย์ขององค์กรได้ว่าใครเป็นผู้กระทำ

4) สภาพพร้อมใช้งาน (Availability) เป็นคุณสมบัติที่เกี่ยวกับการที่สินทรัพย์ขององค์กรที่อยู่ในกระบวนการทางธุรกิจจะต้องสามารถเข้าถึงได้เมื่อจำเป็นเมื่อผู้มีอำนาจเข้าถึงต้องการใช้

โดย M. Schumacher และคณะ [2] ที่นำเสนอในหนังสือแบบรูปความมั่นคงการบูรณาการความมั่นคงและวิศวกรรมระบบ (Security Patterns: Integrating Security and Systems Engineering) ได้กล่าวถึงแบบรูปที่ได้รับความนิยม โดยแบบรูปที่สร้างขึ้นนั้นสามารถนำไปประยุกต์ใช้กับการพัฒนาซอฟต์แวร์ได้ โดยแบ่งออกเป็นทั้งสิ้น 8 กลุ่มรวมทั้งสิ้น 46 แบบรูป โดยกลุ่มของแบบรูปความมั่นคงที่ได้กล่าวโดย M. Schumacher มีดังนี้

1) กลุ่มแบบรูปการจัดการความมั่นคงองค์กรและการจัดการความเสี่ยง (Enterprise Security and Risk Management Patterns)

2) กลุ่มแบบรูปการระบุตัวตนและการพิสูจน์ตัวจริง (Identification & Authentication (I&A) Patterns)

3) กลุ่มแบบรูปแบบจำลองควบคุมการเข้าถึง (Access Control Model Patterns)

4) กลุ่มแบบรูปสถาปัตยกรรมการควบคุมการเข้าถึงระบบ (System Access Control Architecture Patterns)

5) กลุ่มแบบรูปการควบคุมการเข้าถึงระบบการทำงาน (Operating System Access Control Patterns)

6) กลุ่มแบบรูปการตรวจสอบ (Accounting Patterns)

7) กลุ่มแบบรูปสถาปัตยกรรมไฟร์วอลล์ (Firewall Architecture Patterns)

8) กลุ่มแบบรูปความปลอยภัยในการใช้อินเทอร์เน็ต (Secure Internet Applications Patterns)

โดยในงานวิจัยนี้ได้นำแบบรูปที่ได้มาจากงานวิจัย [3] มาเป็นฐานในการวิจัยทั้งสิ้น 4 กลุ่ม จำนวน 20 แบบรูปได้แก่ กลุ่มแบบรูปการจัดการความมั่นคงองค์กรและการจัดการความเสี่ยง กลุ่มแบบรูปการระบุตัวตนและการพิสูจน์ตัวจริง กลุ่มแบบรูปแบบจำลองควบคุมการเข้าถึง และกลุ่มแบบรูปสถาปัตยกรรมไฟร์วอลล์

โดยภายในแบบรูปความมั่งคงของนาย M. Schumacher และคณะ [2] ที่ได้กล่าวไว้ในหนังสือประกอบต่างๆ ไปด้วยส่วนประกอบ ดังนี้

ตารางที่ 2.1 รายละเอียดของแบบรูปความมั่นคงของ M. Schumacher และคณะ [2]

ส่วนประกอบ	รายละเอียดของส่วนประกอบ
ชื่อ (Name)	ชื่อของแบบรูปความมั่นคง โดยในหนังสือเล่มนี้รวมถึงรายละเอียดสั้นๆ ของแบบรูปนี้ด้วย
ชื่อเรียกอื่นของแบบรูป (Also Known As)	ชื่อที่รู้จักใน บางสถานที่ หรือบางเวลาของแบบรูปความมั่นคง

ตารางที่ 2.1 รายละเอียดของแบบรูปความมั่นคงของ M. Schumacher และคณะ [2] (ต่อ)

ส่วนประกอบ	รายละเอียดของส่วนประกอบ
ตัวอย่าง (Example)	ตัวอย่างของปัญหาที่มีการนำแบบรูปความมั่นคงแบบรูปนั้นไปใช้งาน
บริบท (Context)	กล่าวถึงสถานการณ์ที่ควรนำแบบรูปความมั่นคงดังกล่าวไปใช้
ผลเฉลย (Solution)	ผลเฉลยภายใต้แบบรูปความมั่นคง
โครงสร้าง (Structure)	อธิบายรายละเอียดโครงสร้างของแบบรูปความมั่นคง
ไดนามิก (Dynamics)	อธิบายเหตุการณ์ของพฤติกรรมขณะการใช้งานของแบบรูปความมั่นคง
การทำให้เกิดผล (Implementation)	ตัวชี้นำการใช้งานแบบรูปความมั่นคง โดยไม่ได้บังคับให้ทำตามทั้งหมด
ตัวอย่างการแก้ไข (Example Resolved)	ตัวอย่างการแก้ไขปัญหาด้วยแบบรูปความมั่นคง
ส่วนแปรผัน (Variants)	คำอธิบายสั้นๆ ถึงส่วนที่แตกต่าง หรือรายละเอียดพิเศษของแบบรูป
การนำไปใช้ที่ทราบ (Know Uses)	ตัวอย่างการนำไปใช้ของแบบรูปความมั่นคงในระบบที่ใช้งานจริง
ผลที่ได้ (Consequence)	ประโยชน์ที่ได้จากรับและ ของผลเสียที่เป็นไปได้ของแบบรูป
เห็นได้จาก (See Also)	ทำการอ้างถึงแบบรูปความมั่นคงอื่นที่แก้ไขปัญหาเดียวกัน

2.2. งานวิจัยที่เกี่ยวข้อง

2.2.1. การกำหนดความต้องการความมั่นคงโดยใช้ไวยากรณ์ของแบบรูปความมั่นคง (Defining Security Requirements Using Grammar of Security Patterns) โดย กวิน สุภาพร [3]

งานวิจัยนี้ ได้นำเสนอไวยากรณ์ของรูปแบบความมั่นคงที่ได้มาจากการวิเคราะห์แบบรูปความมั่นคง ที่มีการนำเสนอไว้ในหนังสือแบบรูปความมั่นคง การบูรณาการความมั่นคงและวิศวกรรมระบบ (Security Patterns: Integrating Security and Systems Engineering) [2] เป็นจำนวน 20 แบบรูปซึ่งครอบคลุม 4 กลุ่มแบบรูปความมั่นคง ได้แก่ การจัดการความมั่นคง

องค์กรและการจัดการความเสี่ยง การระบุตัวตนและการพิสูจน์ตัวจริง แบบจำลองควบคุมการเข้าถึง และสถาปัตยกรรมไฟล์วอลล์

โดยทำการวิเคราะห์จากส่วนประกอบของแบบรูปที่กล่าวอยู่ในหนังสือแบบรูปความมั่นคง การบูรณาการความมั่นคงและวิศวกรรมระบบ [2] โดยสนใจในส่วนประกอบของแบบรูปดังนี้ "Structure", "Dynamic" "Solution" และ "Example Resolved" โดยมีรายละเอียดคือ

1) โครงสร้าง (Structure) เป็นส่วนประกอบแรกที่ทำการพิจราณาเนื่องมาจากส่วนใหญ่จะแสดงโครงสร้างของแบบรูปด้วยแผนภาพคลาส (Class Diagram) แผนภาพกิจกรรม โดยหากเป็นแผนภาพคลาสจะแสดงให้เห็นถึงข้อมูลสำคัญต่างๆ ของแบบรูป ซึ่งง่ายต่อการพิจารณา และหากเป็นแผนภาพกิจกรรมจะแสดงให้เห็นว่าแบบรูปมีการทำงานอย่างไร ซึ่งส่วนประกอบ "structure" ก็ไม่ได้ถูกแสดงให้เห็นเป็นแผนภาพในทุกๆ แบบรูป ทำให้หากแบบรูปไหนไม่มีในส่วน "structure" ก็ต้องไปพิจราณาจากในส่วนประกอบสำคัญอื่นแทน ได้แก่ "Dynamic" "Solution" และ "Example Resolved"

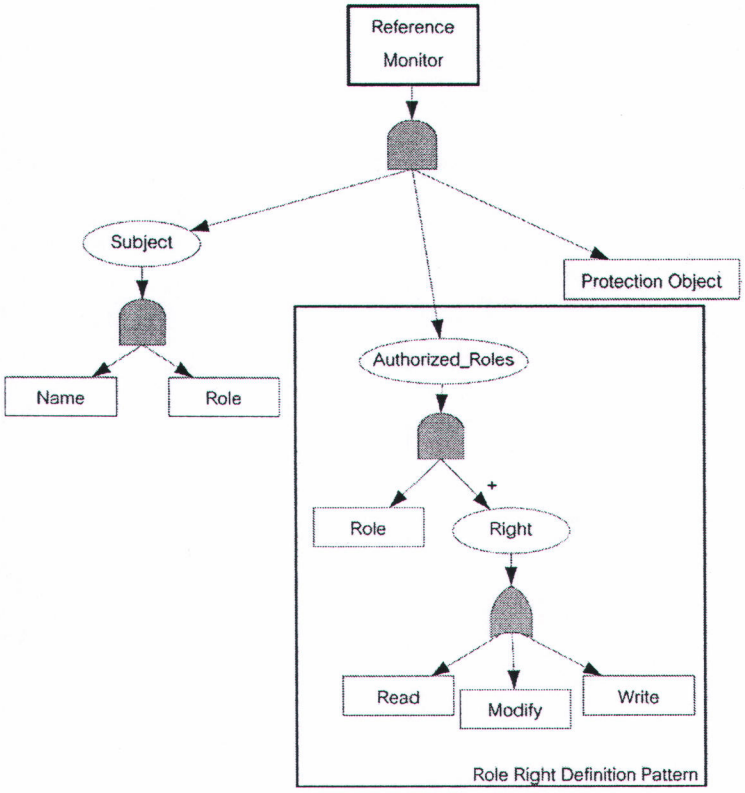
2) พลวัต (Dynamic) เป็นส่วนประกอบที่นำเอาแบบรูปไปประยุกต์ใช้ โดยแสดงเป็นสถานการณ์จำลอง ซึ่งแสดงถึงพฤติกรรมของแบบรูป โดยส่วนใหญ่จะนำเสนอโดยใช้แผนภาพลำดับ (Sequence Diagram) เพื่อแสดงข้อมูลและการติดต่อกันระหว่างองค์ประกอบใดๆ ซึ่งมักพิจารณาควบคู่ไปกับส่วนประกอบ "implementation" เนื่องมาจากนำเสนอแนวทางในการนำแบบรูปไปใช้เช่นกัน

3) ผลเฉลย (Solution) เป็นส่วนประกอบที่นำเสนอผลเฉลยของแบบรูปโดยนำเสนอรายการสิ่งที่จะต้องทำ หรือส่วนประกอบ พร้อมคำอธิบายซึ่งส่วนใหญ่เกี่ยวข้องกับหลักการด้านความมั่นคง (Security Principles)

4) ตัวอย่างการแก้ปัญหา (Example Resolved) เป็นส่วนประกอบที่นำเสนอคุณลักษณะสำคัญ หรือตัวอย่างผลลัพธ์ที่ได้จากการแก้ปัญหา ซึ่งมักอยู่นอกเหนือจากส่วนประกอบ "Structure" "Dynamic" และ "Implementation" แต่ทางผู้วิจัยได้ให้ความสำคัญเป็นลำดับสุดท้ายโดยให้เหตุผลว่า เนื่องมาจากมีความหลากหลายมาก รวมถึงไม่ได้ระบุขอบเขตเงื่อนไขก่อนการใช้แบบรูป

จากส่วนประกอบของแบบรูปดังที่ได้กล่าวไว้ข้างต้นผู้วิจัยได้นำมาวิเคราะห์และสร้างเป็นแผนภาพต้นไม้ความมั่นคง (Security Tree Diagram) เพื่อจะได้นำต้นไม้ความมั่นคงนั้น

ไปแปลงเป็นไวยากรณ์ความมั่นคงไม่พึ่งบริบทต่อไป และนำไวยากรณ์ความมั่นคงไม่พึ่งบริบทไปสร้างเป็นความต้องการความมั่นคง โดยแสดงตัวอย่างต้นไม้ความมั่นคงดังรูปที่ 2.7 และอธิบายรายละเอียดของสัญลักษณ์ที่ปรากฏในแผนภาพต้นไม้ความมั่นคงในตารางที่ 2.2 และแสดงตัวอย่างไวยากรณ์ความมั่นคงไม่พึ่งบริบทในรูปแบบอีบีเอ็นเอฟ [14] รวมถึงผลลัพธ์ดังรูปที่ 2.8



รูปที่ 2.7 ตัวอย่างแผนภาพต้นไม้ความมั่นคง [3]

ตารางที่ 2.2 สัญลักษณ์ ชื่อ และความหมายของสัญลักษณ์ที่ใช้ในแผนภาพต้นไม้ความมั่นคง

สัญลักษณ์	ชื่อ	ความหมาย
	AND gate	ส่วนประกอบบทเครื่องหมาย AND จะต้องประกอบด้วยส่วนประกอบทุกตัวที่ปรากฏภายใต้เครื่องหมาย AND
	OR gate	ส่วนประกอบบทเครื่องหมาย OR จะต้องประกอบด้วยส่วนประกอบบางตัวที่ปรากฏภายใต้เครื่องหมาย OR
	PLUS	แสดงความสัมพันธ์ แบบ 0...* (ภายใต้ OR gate)
		แสดงความสัมพันธ์ แบบ 1...* (ภายใต้ AND gate)
	Non-Terminal	แสดงส่วนประกอบที่ประกอบด้วยองค์ประกอบย่อยอื่นๆ
	Terminal	แสดงส่วนประกอบที่ทราบค่า หรือไม่สามารถแยกเป็นองค์ประกอบย่อยอื่นได้อีก

Ref-Monitor	=	Subject , Authorized-Roles , Protection-Object , "." ;
Subject	=	Subject-Name , ", who acquires", Role-Name , " role, " ;
Subject-Name	=	? The name of subject such as person or process ? ;
Role-Name	=	? The defined role in organization based on its policy ? ;
Authorized-Roles	=	"is authorized to" , Right-List ;
Right-List	=	Right , {"," Right} ;
Right	=	["read" | "write" | "modify" | User-Define-Right] ;
		(* users can define a new right by themselves.
		This feature is supported by the prototyping tool *)
User-Define-Right	=	? A new right which defined by user ? ;
Protection-Object	=	? The name of asset which subject attempt to access ? ;

ตัวอย่างผลลัพธ์

Somsak, who acquires doctor role, is authorized to read, modify patient records.

Somsri, who acquires nurse role, is authorized to read the medical orders.

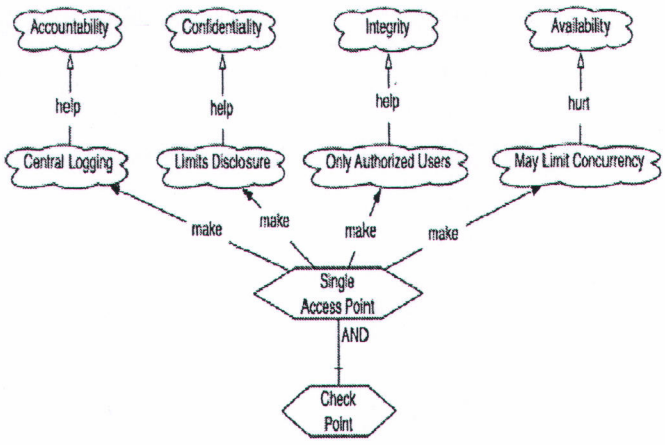
รูปที่ 2.8 ตัวอย่างไวยากรณ์ความมั่นคงไม่พึ่งบริบทและผลลัพธ์ [3]

สิ่งที่นำมาพิจารณาใช้จากงานวิจัยนี้ คือ ไวยากรณ์ความมั่นคงไม่พึ่งบริบท และความสัมพันธ์ระหว่างแบบรูปความมั่นคงกับความต้องการความมั่นคง โดยนำไวยากรณ์มาเพื่อช่วยสร้างความต้องการความมั่นคงให้ถูกต้อง ชัดเจน และลดความกำกวม และความสัมพันธ์ระหว่างแบบรูปความมั่นคงกับความต้องการความมั่นคงมาช่วยในการระบุความเกี่ยวเนื่องกันระหว่างแบบรูปความมั่นคงกับความต้องการความมั่นคง

2.2.2. การเลือกแบบรูปความมั่นคงเพื่อเติมเต็มความต้องการความมั่นคง (Selecting Security Patterns that Fulfill Security Requirements) โดย M. Weiss และ H. Mouratidis [6]

งานวิจัยนี้ ได้กล่าวถึงการเพิ่มขึ้นเป็นจำนวนมากของแบบรูปความมั่นคงทำให้เกิดปัญหาในการเลือกแบบรูปเพื่อใช้ในความมั่นคงที่แตกต่างกัน โดยได้กล่าวถึงเกณฑ์การเลือกแบบรูปความมั่นคงให้กับความต้องการความมั่นคง

โดยการใช้ Goal-Oriented Requirements Language (GRL) ในการแสดงให้เห็นถึงความต้องการที่ไม่ใช่หน้าที่ ให้เป็นทางการและชัดเจนยิ่งขึ้น โดยการแปลงความต้องการความมั่นคงให้กลายเป็นแบบจำลอง GRL ดังรูปที่ 2.9 โดยสัญลักษณ์ของแบบจำลองแสดงดังตารางที่ 2.3



รูปที่ 2.9 แบบจำลอง GRL ของ "Single Access Point" [6]

ตารางที่ 2.3 สัญลักษณ์ของแบบจำลอง GRL

สัญลักษณ์	คำอธิบาย
	แบบรูปที่ใช้
	ปัจจัยเป้าหมายอ่อน (soft-goal elements)
	หรือ
	และ

โดยเมื่อทำการแจกแจงออกมาเสร็จเขานำมาจัดเก็บอยู่ในรูปแบบดังนี้

pattern(Name, FulfilledNFRs, RequiredNFRs)

หากพิจารณาแบบรูปความมั่นคงสำหรับการควบคุมการเข้าถึงได้แก่ “Single Access Point” “check Point” “Security Session” และ “Role-Based Access Control (RBAC)”จะสามารถนำเสนอแบบรูปได้ดังต่อไปนี้

pattern('Single Access Point', ['Integrity', 'Confidentiality', 'Accountability'], ['Availability']).

pattern('Check Point', ['Availability', 'Integrity', 'Confidentiality'], []).

pattern('Security Session', ['Availability', 'Integrity', 'Confidentiality', 'Accountability', 'Usability'], []).

pattern('RBAC', ['Manageability', 'Availability', 'Integrity', 'Confidentiality'], []).

เมื่อได้ความสัมพันธ์ระหว่างแบบรูปแล้วก็มีการเรียกใช้ดังนี้

uses(and, 'Single Access Point', 'Check Point').

uses(or, 'Check Point', 'Security Session').

uses(or, 'Check Point', 'RBAC').

โดยเมื่อมีการระบุถึงการเรียกใช้ในแบบต่างๆ ก็จะเพื่อนำไปคำนวณระดับความพอใจเพื่อนำมาใช้ในการเลือกแบบรูปความมั่นคงที่เข้ากับความต้องการที่ระบุมา

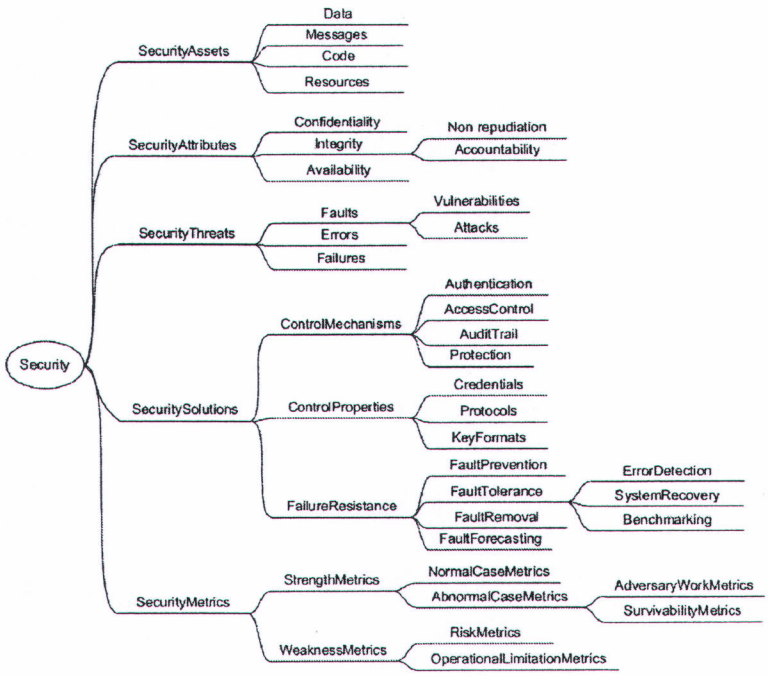
สิ่งที่นำมาพิจารณาใช้จากงานวิจัยนี้ คือ ทำให้เล็งเห็นถึงปัญหาในการเลือกแบบรูปความมั่นคงให้เข้ากับในแต่ละโครงการ ซึ่งต้องการที่จะระบุคุณสมบัติหลักของแบบรูปความมั่นคงจากความต้องการซึ่งเป็นความต้องการที่ไม่ใช่หน้าที่ ซึ่งยากต่อการพิจราณาในทุกๆ โครงการ โดยในงานนี้ไม่ได้กล่าวถึงการนำกลับมาใช้ใหม่จากโครงการในอดีต

2.2.3. **การแบ่งประเภทของความมั่นคงข้อมูลสำหรับระบบบริการแบบศูนย์กลาง** (A Taxonomy of Information Security for Service-Centric Systems) **โดย** Pekka Savolainen **และคณะ** [15]

งานวิจัยนี้ ได้เล็งเห็นถึงการติดต่อกันอย่างกว้างขวาง และการเจริญเติบโตอย่างรวดเร็วของอินเทอร์เน็ต ซึ่งจำเป็นการกระตุ้นที่ต้องการความมั่นใจและความมั่นคงของข้อมูลในอินเตอร์เน็ตที่เพิ่มมากยิ่งขึ้นเรื่อยๆ แต่การรวมกันของ ส่วนประกอบ และบริการที่เกิดจากหลากหลายแหล่ง ซึ่งปราศจากวิธีการรับรองคุณภาพอาจทำให้เกิดจุดอ่อนที่เกิดขึ้นกับระบบทำให้เกิดการโจมตีที่มุ่งร้ายเกิดขึ้น เพื่อที่จะประกันถึงความมั่นคงของระบบงานวิจัยนี้ต้องการนำเสนอความเข้าใจในทางเดียวกันเกี่ยวกับความมั่นคงและการวัดมันได้อย่างไร โดยในงานนำเสนอชิ้นนี้ได้นำเสนอการแบ่งประเภทของความมั่นคงข้อมูลโดยอยากให้ใช้ร่วมกับสถาปัตยกรรมซอฟต์แวร์ของระบบบริการแบบศูนย์กลาง

การแบ่งประเภทของความมั่นคงข้อมูลสำหรับสถาปัตยกรรมเชิงบริการ (Service-oriented architectures: SOA) โดยแบ่งออกได้ 5 ด้าน ได้แก่ สินทรัพย์ความมั่นคง (Security assets) คุณลักษณะความมั่นคง (Security attributes) ภัยคุกคามความมั่นคง (Security

threats), ผลเฉลยความมั่นคง (Security solutions) และการวัดความมั่นคง (Security metrics) โดยแสดงดังรูปที่ 2.10



รูปที่ 2.10 การแบ่งประเภทของความมั่นคงข้อมูลสถาปัตยกรรมเชิงบริการ [15]

1) สินทรัพย์ความมั่นคง หมายถึงทุกๆ ประเภทของสินทรัพย์ที่ต้องการป้องกันและวัดค่า เช่น ข้อมูล (data) ข้อความ (messages) คำสั่ง (code) ทรัพยากร (resources) เป็นต้น
2) คุณลักษณะความมั่นคง หมายถึงคุณลักษณะของระบบที่ต้องการทำการปรับปรุงใหม่ได้แก่
   (1) ความเป็นความลับ คือการป้องกันผู้ที่ไม่ได้รับสิทธิ์จากสินทรัพย์นั้นๆ
   (2) บูรณภาพ คือการป้องกันผู้ที่ไม่ได้รับสิทธิ์ในการแก้ไขหรือลบทิ้ง
   (3) สภาพพร้อมใช้งาน คือการป้องกันผู้ที่ไม่ได้รับสิทธิ์มาขัดขวางข้อมูล
3) ภัยคุกคามความมั่นคง หมายถึงทุกสิ่งที่จะทำให้ระบบเกิดความผิดพลาดได้แก่
   (1) ความผิดพลาด (Faults) คือสาเหตุที่ทำให้เกิดข้อผิดพลาด
   (2) ข้อผิดพลาด (Error) คือความคลาดเคลื่อนจากสภาพภายนอกระบบ
   (3) ความล้มเหลว (Failures) คือพฤติกรรมของตัวบริการที่ผิดเพี้ยนไปจากการทำงานที่ถูกต้อง

4) ผลเฉลยความมั่นคง หมายถึงผลเฉลยที่จะทำให้ไม่เกิดความผิดพลาดขึ้นอีกโดยจะขึ้นอยู่กับโครงสร้างและพฤติกรรมของระบบ โดยในผลเฉลยความมั่นคงแบ่งออกเป็นกลุ่มได้เป็น 3 ชนิด ได้แก่ กลไกการควบคุม (Control mechanisms) คุณสมบัติการควบคุม (Control properties) และการทนทานความล้มเหลว (Failure resistance)

5) การวัดความมั่นคง เป็นส่วนสำคัญในการวัดส่วนที่ดีของระบบเป้าหมายถึงมาตรการการตอบโต้สำหรับภัยคุกคามความมั่นคง

สิ่งที่นำมาพิจารณาใช้จากงานวิจัยนี้ คือโครงสร้างของความมั่นคงและการแบ่งประเภทของความมั่นคงข้อมูล ซึ่งนำมาช่วยขยายรายละเอียดในการจัดเก็บและค้นคืนในงานวิจัยนี้

2.2.4. **การกู้คืนเส้นเชื่อมเพื่อตามรอยในระบบจัดการสินทรัพย์ของซอฟต์แวร์โดยการใช้วิธีการทางการค้นคืนสารสนเทศ** (Recovering Traceability Links in Software Artifact Management Systems using Information Retrieval Methods) **โดย** Andrea De Lucia **และ คณะ** [16]

ในงานวิจัยนี้ เริ่มที่ผู้วิจัยเล็งเห็นถึงความเป็นไปได้ที่จะนำเอาการค้นคืนสารสนเทศมาช่วยในการจัดการสินทรัพย์โดยการนำเอาการค้นคืนสารสนเทศมาช่วยในการตามรอยของสินทรัพย์ที่มีอยู่ในระบบ โดยทำการเทียบความคล้ายคลึงระหว่างสินทรัพย์ที่ได้ทำการระบุโดยผู้ใช้ว่าสินทรัพย์ใดมีความใกล้เคียงกันมากก็นำเสนอในลำดับต้นๆ โดยได้มีการพัฒนาระบบการจัดการสินทรัพย์ที่มีชื่อว่า “Advanced Artifact Management System: ADAMS”

โดยงานวิจัยนี้ได้ทำการศึกษาและการนำเสนอผลการวิจัยอย่างต่อเนื่องจนถึงปัจจุบัน โดยนำเสนอถึงวิธีในการค้นคืนอย่างมีประสิทธิภาพ หรือตัวระบบที่มีการปรับเปลี่ยนไปให้ทำงานได้ดียิ่งขึ้น

สิ่งที่นำมาพิจารณาใช้จากงานวิจัยนี้ คือแนวความคิดในการใช้การค้นคืนสารสนเทศมาใช้ในการช่วยในการค้นคืนความต้องการความมั่นคงและแบบรูปความมั่นคง

ในบทนี้ผู้วิจัยได้ทำการรวบรวมและศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องต่างๆ เพื่อนำองค์ความรู้ต่างๆ ที่ได้ผ่านการศึกษามาทำการวิเคราะห์และออกแบบระบบ โดยจะกล่าวต่อต่อไปในบทที่ 3