

บทที่ 6

การจัดทำข้อกำหนดความต้องการ และเอกสารความต้องการ
(Requirements specification and documentation)



เนื้อหา

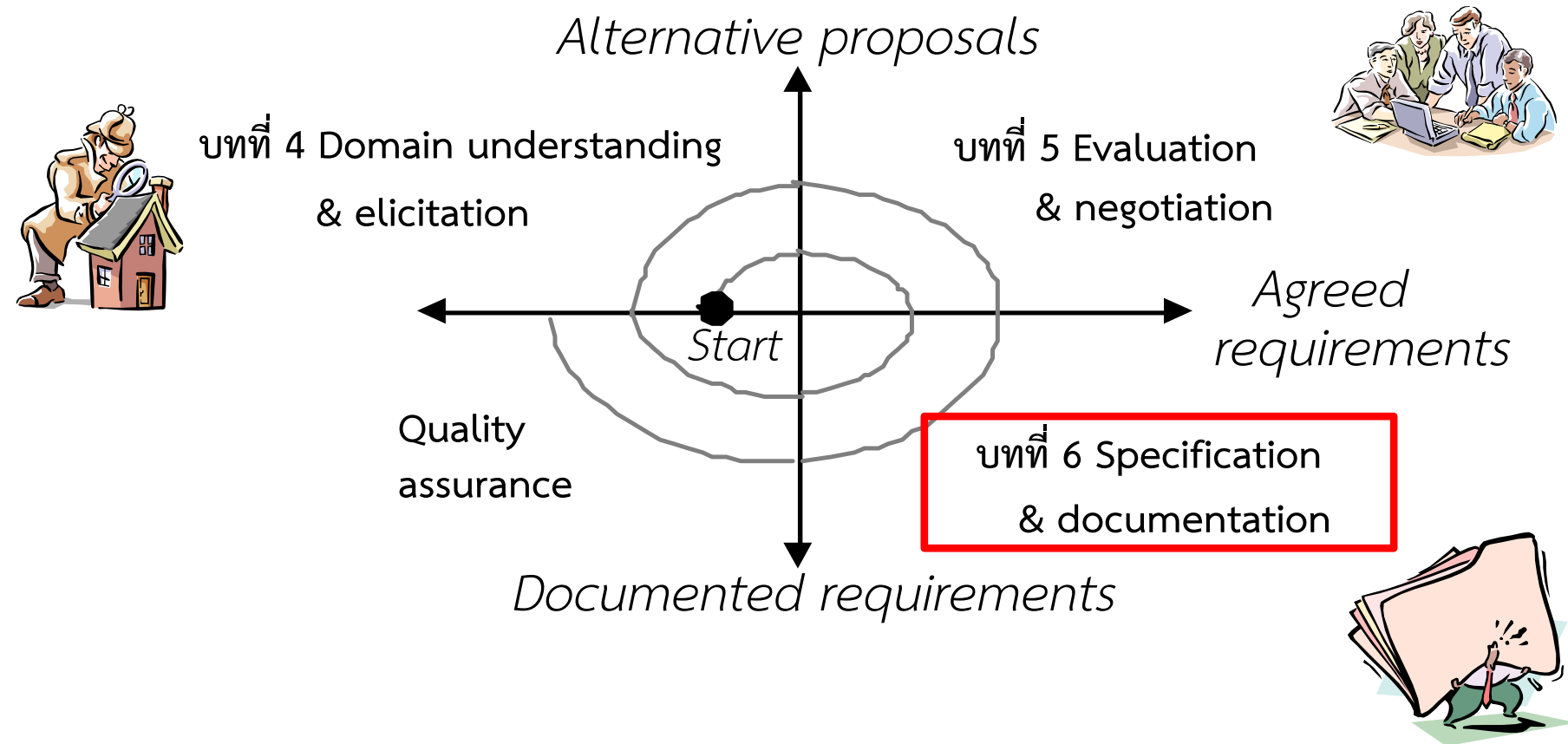
- บริบทความต้องการ
- การเขียนความต้องการ
- การใช้สัญลักษณ์แผนภาพ
- การจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ

วัตถุประสงค์การเรียนรู้

- เพื่อให้เข้าใจเกี่ยวกับเอกสารความต้องการ
- เพื่อให้รู้จักองค์ประกอบต่างๆ ของเอกสารความต้องการ
- เพื่อให้มีความรู้เกี่ยวกับแม่แบบการเขียนเอกสารความต้องการ
- เพื่อให้เข้าใจในการเขียนข้อกำหนดความต้องการแบบเป็นทางการ

ภาพที่ 6-1 Spiral model

- กล่าวเกี่ยวกับกระบวนการวิศวกรรมความต้องการ (RE process)



บริบทของความต้องการ

(Requirements in context)

บริบทของความต้องการ

- การเริ่มทำเอกสารความต้องการ (Requirements Document: RD) ต้องทำความเข้าใจความต้องการให้ชัดเจน จึงจะสามารถสร้างเอกสารความต้องการที่ดีได้
- บทนี้จะกล่าวถึงองค์ประกอบหลักในการสร้าง RD ได้แก่
 - เงื่อนไขของระบบ
 - ขอบเขตของระบบ
 - คุณสมบัติของความต้องการ
 - การติดตามความต้องการ

ความหมายของเงื่อนไข

- สิ่งที่ใช้ควบคุมลักษณะการทำงานของระบบ เพื่อไม่ให้ทำงานนอกเหนือจากสิ่งที่กำหนดไว้
- ตัวอย่างเช่น
 - ▣ งบประมาณ
 - ▣ ระยะเวลาในการทำงาน

ประเภทของเงื่อนไข

- เงื่อนไขด้านความปลอดภัย (Safety constraints)
- เงื่อนไขด้านสมรรถนะของระบบ (Performance constraints)
- เงื่อนไขด้านงบประมาณ (Budget constraints)
- เงื่อนไขด้านประสิทธิภาพที่ส่งผลต่อการทำงานทั้งระบบ (Performance constraints often apply across whole systems)
- เงื่อนไขที่ครอบคลุมระบบ และครอบคลุมเฉพาะส่วน (Global and local constraints)

เงื่อนไขด้านความปลอดภัย

- เงื่อนไขทางด้านความปลอดภัยถือว่าเป็นเงื่อนไขที่สำคัญ
- ตัวอย่างเช่น
 - ระบบรักษาความปลอดภัยของเครือข่ายอินเทอร์เน็ต
 - ไฟร์วอลล์ (Firewall) ทำหน้าที่ป้องกันการโจมตีจากภายนอก เพื่อไม่ให้เข้าถึงช่องโหว่ที่อาจเกิดขึ้นบนเครื่องแม่ข่าย และป้องกันไม่ให้ข้อมูลรั่วไหลถึงบุคคลภายนอก
 - มีระบบ Authentication เพื่อพิสูจน์ตัวตน

เงื่อนไขด้านสมรรถนะของระบบ

- ระบบที่ดีมักขึ้นอยู่กับ การออกแบบระบบ
- การวัดประสิทธิภาพการทำงานของระบบจะวัดจากความสามารถที่ระบบทำได้
- ตัวอย่างเช่น
 - “ระบบลงทะเบียนเรียนสามารถรองรับผู้ใช้งานได้จำนวนมาก และตลอดเวลา”

เงื่อนไขด้านงบประมาณ

- งบประมาณเป็นอีกหนึ่งปัจจัยที่เป็นแรงผลักดันให้เกิดโครงการการพัฒนาทางสารสนเทศ
- การประมาณการงบประมาณจะต้องมีการติดตามการใช้งบประมาณ
- เงื่อนไขและการออกแบบมีผลต่องบประมาณ
- ตัวอย่างเช่น
 - “วิศวกรซอฟต์แวร์ต้องทำงานได้ภายใต้งบประมาณที่จำกัด”

เงื่อนไขด้านประสิทธิภาพที่ส่งผลต่อการทำงานทั้งระบบ

- เป็นเงื่อนไขที่ส่งผลกับการทำงานของระบบทั้งหมด ซึ่งจะครอบคลุมเรื่องต่าง ๆ ได้แก่
 - ▣ ค่าใช้จ่าย
 - ▣ ความน่าเชื่อถือ
 - ▣ การดูแลบำรุงรักษา

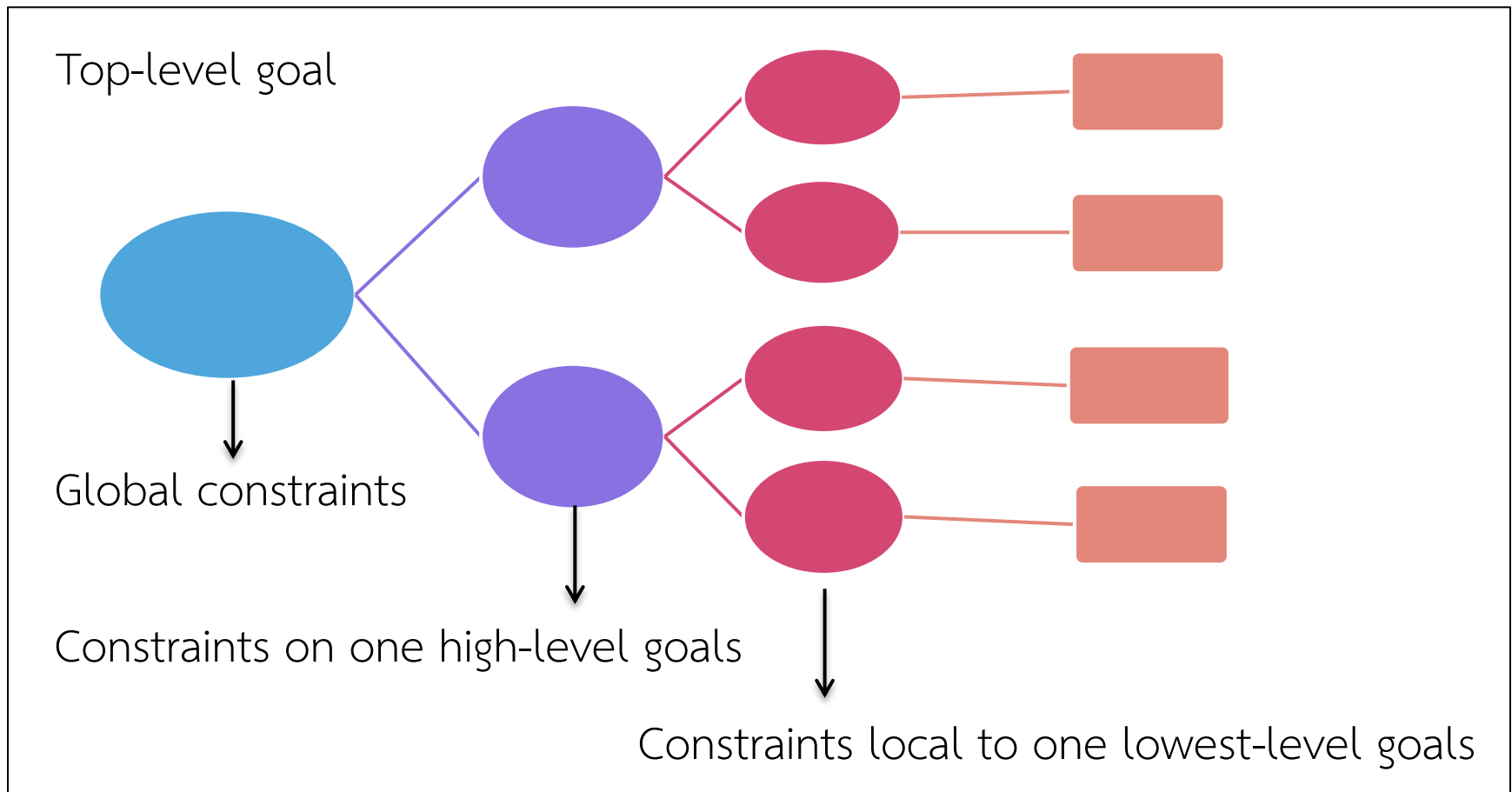
เงื่อนไขที่ครอบคลุมระบบ และครอบคลุมเฉพาะส่วน

- เงื่อนไขครอบคลุมทุกส่วนของระบบ (Constraints is global)
- เงื่อนไขครอบคลุมเฉพาะส่วนในระดับสูงของระบบ (Constraint is local to a high-level)
- เงื่อนไขครอบคลุมเฉพาะส่วนในระดับต่ำที่สุดของระบบ (Constraint is local to a lowest-level)

ตารางที่ 6-1 ชนิดของเงื่อนไข

ครอบคลุมทุกส่วน ของระบบ	ครอบคลุมเฉพาะส่วน ในระดับสูง	ครอบคลุมเฉพาะส่วน ในระดับต่ำสุด
<p>- การเขียนระบบทั้งระบบ ต้องกำหนดขอบเขตของ การพัฒนาระบบโดย ใช้เวลา 6 เดือน</p>	<p>- การเขียนมอดูลของระบบ ต้องกำหนดขอบเขตของ การพัฒนาแต่ละมอดูลโดย ใช้เวลา 4 สัปดาห์</p>	<p>- การเขียนฟังก์ชันของระบบ ต้องกำหนดขอบเขตของ การพัฒนาแต่ละฟังก์ชันโดย ใช้เวลา 3 วัน</p>

ภาพที่ 6-2 ชนิดของเงื่อนไข



การกำหนดขอบเขต

- สิ่งสำคัญที่จะทำให้โครงการประสบผลสำเร็จ คือ ต้องกำหนดสิ่งที่ระบบสามารถทำงานได้อย่างถูกต้อง และครอบคลุม
- สิ่งนี้เรียกว่า “ขอบเขตของระบบ”

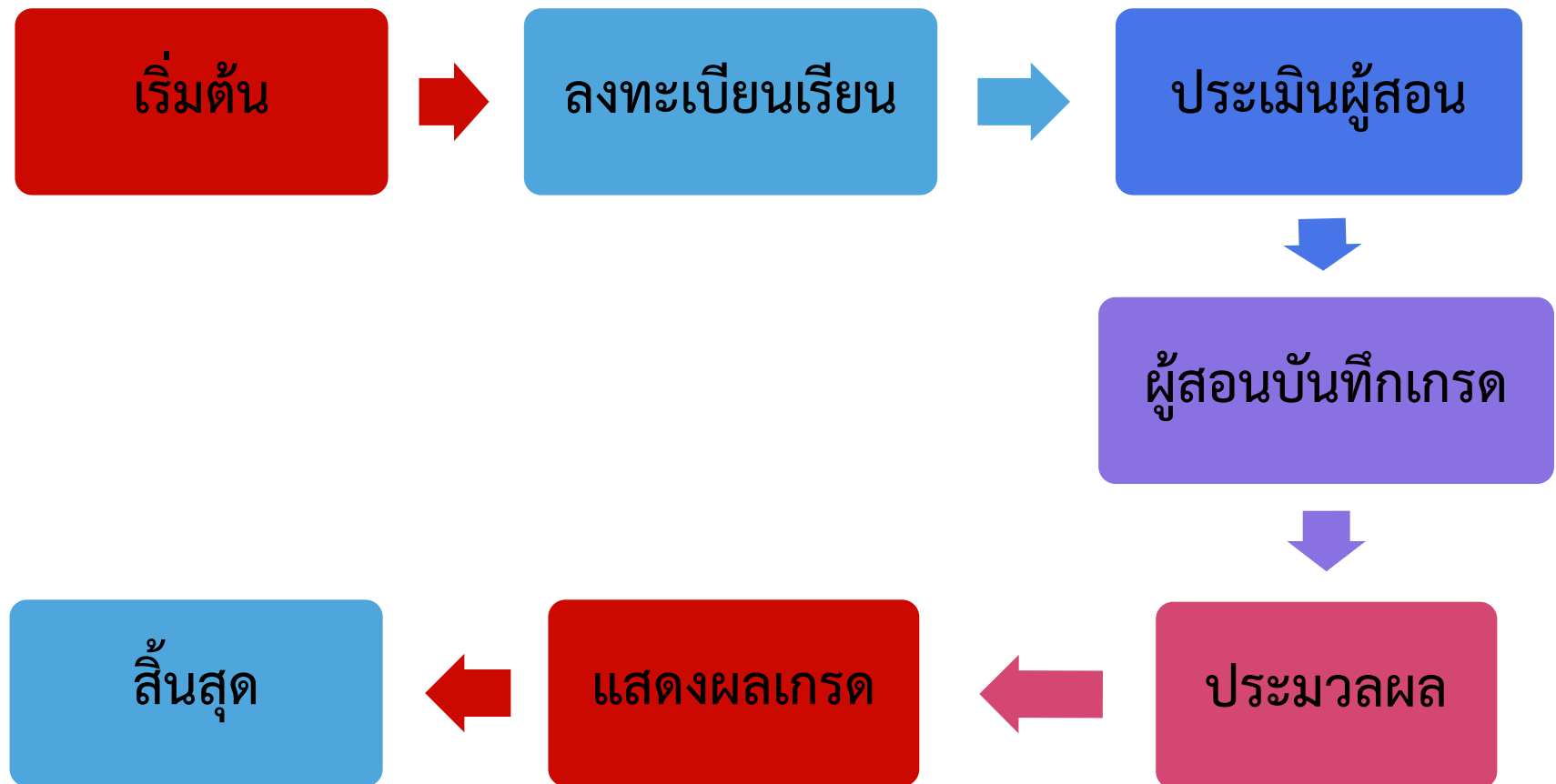
ขอบเขตถูกกำหนดโดยการเจรจาต่อรอง

- คือ สิ่งที่ใช้และวิศวกรซอฟต์แวร์เข้าใจตรงกัน
- ตัวอย่างเช่น
 - ความต้องการของผู้ใช้
 - ราคาของซอฟต์แวร์
 - ระยะเวลาในการดำเนินงาน และวันที่ส่งมอบงาน

การระบุลำดับความสำคัญ

- ต้องมีการจัดลำดับความสำคัญของการทำงาน สิ่งใดควรทำก่อน สิ่งใดควรทำภายหลัง

ภาพที่ 6-3 ตัวอย่างแผนภาพการทำงานการดูแลการเรียน



ข้อสำคัญที่จะทำให้งานประสบความสำเร็จ

- กำหนดราคาของซอฟต์แวร์อย่างสมเหตุสมผล
- จัดลำดับความสำคัญของความต้องการของผู้ใช้

เงื่อนไขด้านงบประมาณที่มีความเห็นร่วมกัน

- ระบบควรมีการจำกัดขอบเขตของงบประมาณ เพราะบางครั้งความต้องการของผู้ใช้อาจมีมากเกินไปงบประมาณที่กำหนดไว้
- ออกแบบการทำงานที่เป็นไปได้ภายใต้งบประมาณที่กำหนด
- อาจไม่สามารถปรับเปลี่ยนความต้องการบางรายการได้ เนื่องจากมีค่าใช้จ่ายสูงเกินไปและทำให้การส่งงานล่าช้ากว่ากำหนด

คุณสมบัติที่ดีของความต้องการ

- ความต้องการต้องเป็นมากกว่าเอกสารหรือข้อความ
- ความต้องการต้องได้รับการตรวจสอบและจัดลำดับความสำคัญ
- อย่ามองว่าความต้องการเป็นเพียงกระดาษที่เก็บไว้ในตู้เอกสาร
- แต่ต้องมองเป็น “แผนที่การนำทาง”

การใช้เช็คลิสต์ในการตรวจสอบ (Check list)

- เป็นแบบฟอร์มการตรวจสอบ
- ใช้ในการตรวจสอบระบบ
- ใช้ในการตรวจสอบเอกสารความต้องการ
- ต้องสามารถระบุงานก่อนหลังได้



การติดตามความต้องการ

- การจัดทำเอกสารความต้องการ วิศวกรซอฟต์แวร์กับผู้ใช้ต้องมีการตกลงหาข้อยุติร่วมกัน
- การเปลี่ยนแปลงกำหนดการหรือคนในทีมลาออก อาจเป็นปัจจัยที่ทำให้โครงการล้มเหลว
- การติดตามการเปลี่ยนแปลงความต้องการในการออกแบบ เพื่อให้ตอบสนองความต้องการของผู้ใช้
- เครื่องมือความต้องการจะสามารถตรวจสอบการออกแบบได้ แต่ไม่สามารถตรวจสอบได้ว่าใครเป็นผู้ออกแบบ

การเขียนความต้องการ (Requirements writing)

ความต้องการที่ไม่สมบูรณ์

- เกิดขึ้นก็ต่อเมื่อ
 - ไม่ทราบสิ่งที่ผู้ใช้ต้องการอย่างแท้จริง
 - เขียนความต้องการไม่ชัดเจน
 - เขียนความต้องการที่สมบูรณ์แบบจนมากเกินไป

การร่างและการปรับปรุง

- เขียนความต้องการให้มีความชัดเจนมากที่สุด
- ให้ร่างความต้องการแบบคร่าว ๆ ก่อน แล้วจึงปรับปรุงให้ตรงกับความต้องการของผู้ใช้
- เปรียบเทียบกับผู้ใช้อีกครั้งว่าต้องการความต้องการนี้จริงหรือไม่ และจดบันทึกอย่างเป็นลายลักษณ์อักษร เพื่อเป็นข้อตกลงระหว่างผู้ใช้กับวิศวกรซอฟต์แวร์

โครงสร้างของความต้องการที่ดี

- ต้องระบุผู้มีส่วนได้ส่วนเสีย (Stakeholders)
- มีการกำหนดจุดที่ผู้ใช้พึงพอใจในแต่ละขั้นตอน และให้ผู้ใช้ตรวจสอบว่าตรงตามที่ต้องการหรือไม่
- มีส่วนที่เป็นเครื่องมือในการทดสอบซอฟต์แวร์ (Test case) เพื่อตรวจสอบว่าเป็นไปตามความต้องการหรือไม่

วิธีการเขียนความต้องการให้มีคุณภาพ

- ความต้องการควรเขียนด้วยประโยคที่เรียบง่าย เป็นไปได้ และกระชับ แต่ไม่สั้นจนเกินไป หรือไม่สามารถจับใจความได้
 - หลีกเลี่ยงการใช้ศัพท์เทคนิค ภาษาต่างประเทศที่เข้าใจได้ยาก ตัวอักษรย่อ หรือ ศัพท์แสง หากจำเป็นต้องใช้ ควรอธิบายให้ชัดเจนว่าต้องการจะสื่อความหมายว่าอย่างไร
 - ระบุให้ชัดเจนว่าผู้มีส่วนได้ส่วนเสียกับความต้องการนั้น ๆ เป็นใคร
 - ต้องสามารถตรวจสอบได้ว่าความต้องการนั้น ๆ ทำได้จริงหรือไม่

รูปแบบที่ไม่เหมาะสมของการเขียนความต้องการ

- หลีกเลียงคำกำกวม พยายามเขียนความต้องการให้ชัดเจน
 - คำที่ควรระวัง ได้แก่ “หรือ”
- ไม่ควรเขียนความต้องการซ้อนทับกัน
 - คำที่ควรระวัง ได้แก่ “และ หรือ กับ รวมถึง”
- ไม่ควรคาดการณ์ความต้องการล่วงหน้า
 - คำที่ควรระวัง ได้แก่ “มักจะ โดยทั่วไป ตามปกติ เป็นปกติ”

รูปแบบที่ไม่เหมาะสมของการเขียนความต้องการ (2)

- ไม่เขียนสิ่งที่ไม่สามารถวัดได้
 - คำที่ควรระวัง ได้แก่ “เป็นมิตรกับผู้ใช้ อเนกประสงค์ มีประสิทธิภาพสูง”
- ไม่ควรเขียนความต้องการที่มีขอบเขตกว้างเกินไป จนไม่สามารถตัดสินใจได้ว่าจะเลือกทางใด
 - คำที่ควรระวัง ได้แก่ “ถ้า เมื่อ แต่ ยกเว้น เว้นแต่ แม้ว่า”

รูปแบบที่ไม่เหมาะสมของการเขียนความต้องการ (3)

- หลีกเลี่ยงการทำเอกสารความต้องการควบคู่หรือรวมกับการทำแผนงาน
- แผนงานต้องได้รับการปรับปรุงแก้ไขตลอดระยะเวลาของโครงการ
- ความต้องการต้องมีเสถียรภาพ
- เมื่อนำเอกสารความต้องการ และแผนงานมารวมกันจะทำให้เพิ่มจำนวนงานที่ต้องทำมากขึ้น

ข้อมูลนำเข้า

- ข้อมูลนำเข้า คือ นิยามต่าง ๆ ที่ชัดเจน และเกี่ยวข้องกับคุณสมบัติ (Features) ทั้งหมดของระบบ

ข้อมูลนำเข้า (2)

- โดยผู้มีส่วนได้ส่วนเสียเห็นพ้องต้องกัน ได้แก่
 - วัตถุประสงค์โดยทั่วไป (General objectives)
 - ความต้องการของระบบ (System requirements)
 - ความต้องการของซอฟต์แวร์ (Software requirements)
 - สมมติฐานต่าง ๆ ที่เกี่ยวกับสิ่งแวดล้อม (Environment assumptions)
 - คุณสมบัติของโดเมนที่เกี่ยวข้อง (Domain properties)
 - ข้อกำหนดของแนวคิด (Concept definitions)
- โดยทั้งหมดที่กล่าวมาจะเป็นข้อมูลที่เกี่ยวข้องกับ System-to-be และในบางครั้งอาจเกิดขึ้นจาก System-as-is

ข้อมูลส่งออก

- ข้อมูลส่งออก คือ ข้อมูลที่ได้จากขั้นตอนการจัดทำเอกสารความต้องการ และเป็นเอกสารความต้องการเวอร์ชันที่ 1 (First version of requirements document)
- เอกสาร RD จะครอบคลุมถึงข้อกำหนดความต้องการที่มีความชัดเจน โครงสร้างที่ถูกจัดการให้สอดคล้อง โดยจะต้องคำนึงถึงคุณภาพที่กล่าวไว้ในบทที่ 1

ข้อมูลส่งออก (2)

- โครงสร้างของเอกสาร RD ที่ดีจะต้องประกอบไปด้วยคุณสมบัติ ได้แก่
 - สามารถเข้าใจได้ง่าย
 - มีข้อกำหนดความต้องการที่ชัดเจน
 - ใช้ภาษาที่เหมาะสม โดยภาษาจะต้องทำให้การสื่อสารระหว่างผู้มีส่วนได้ส่วนเสีย และวิศวกรซอฟต์แวร์มีความเข้าใจตรงกัน
 - สามารถดึงเอาข้อมูลเหล่านั้นมาวิเคราะห์หาสาเหตุ เพื่อปรับเปลี่ยนตามความเหมาะสม

ข้อมูลส่งออก (3)

- ส่วนหนึ่งใช้ในการวิเคราะห์ข้อกำหนด เพื่อใช้ในกระบวนการตรวจสอบ (Validation) และการทวนสอบ (Verification) ซึ่งเป็นกระบวนการ RE ขั้นต่อไป

การจัดทำเอกสารความต้องการด้วยภาษาธรรมชาติ แบบไม่มีโครงสร้าง

- ข้อความที่เป็นภาษาธรรมชาติจะสามารถเข้าใจได้ง่าย และครอบคลุมทุกส่วน โดยไม่ต้องอาศัยการฝึกฝนอื่น ๆ
- ข้อความที่เป็นภาษาที่มนุษย์ใช้สื่อสารในชีวิตประจำวัน ซึ่งสามารถเข้าใจได้ง่ายและครอบคลุมทุกส่วน โดยไม่ต้องอาศัยการฝึกฝนอื่น ๆ

ข้อผิดพลาดของภาษาธรรมชาติ

- มีความกำกวม (Ambiguities)
- มีข้อมูลที่ไม่เกี่ยวข้อง (Noise)
- ใช้ข้อความที่ยังไม่ถูกนิยามมาก่อน (Forward references)
- มีข้อความกำหนดในเวลาที่ยาวเกินไป (Remorse)
- ไม่สามารถวัดในเชิงเปรียบเทียบได้ (Unmeasurable statement)
- ไม่ชัดเจน (Opacity)

การจัดทำเอกสารความต้องการด้วยภาษาธรรมชาติ

แบบมีโครงสร้าง

- ปัญหาที่เกิดขึ้นในการจัดทำเอกสารความต้องการด้วยภาษาธรรมชาติแบบมีโครงสร้าง ทำให้เกิดการจัดทำเอกสาร 2 แบบ ได้แก่
 - การเขียนโดยใช้ภาษาธรรมชาติที่เป็นมาตรฐานเฉพาะพื้นที่ (Local rules)
 - การเขียนโดยใช้ภาษาธรรมชาติที่สอดคล้องกับมาตรฐานระดับสากล (Global rules)

การเขียนโดยใช้ภาษาธรรมชาติที่เป็นมาตรฐานเฉพาะที่

- มีเทคนิคที่เป็นมาตรฐานเฉพาะที่ ได้แก่
 - การใช้กฎของรูปแบบสำหรับเงื่อนไขของภาษาธรรมชาติ
(Using stylistic rules for natural language specification)
 - ใช้ตารางการตัดสินใจในเงื่อนไขที่ซับซ้อน
(Use decision tables for complex combinations of conditions)
 - การใช้เทมเพลตที่กำหนดไว้ (Using predefined statement templates)

การใช้กฎของรูปแบบสำหรับเงื่อนไขของภาษาธรรมชาติ

- หลักมาตรฐานของการเขียนอาจปรับเปลี่ยนเนื้อหาตามบริบทของการเขียนเอกสารความต้องการ ได้แก่
 - ระบุว่าใครเป็นผู้อ่านและเขียนเนื้อหาให้สอดคล้อง
 - ต้องสามารถบอกได้ว่างานก่อนหน้านี้ใครเป็นผู้กระทำ
 - สิ่งแรก คือ ที่มาและความสำคัญ
 - สิ่งสุดท้าย คือ การสรุปความ
 - ตรวจสอบให้แน่ใจว่าทุกแนวคิดถูกกำหนดไว้ก่อนนำมาใช้

การใช้กฎของรูปแบบสำหรับเงื่อนไขของภาษาธรรมชาติ (2)

- ประโยคนั้น ๆ ควรสั้นกระชับ และกะทัดรัด
- ในแต่ละชุดคำสั่งจะต้องกำหนดด้วยว่าทำอะไรได้บ้าง
- ใช้การยกตัวอย่างให้เข้าใจได้ง่าย
- ใช้ข้อความอธิบายประกอบ Diagrams เพื่อให้เกิดความเข้าใจ

การใช้เทมเพลตในการนำเสนอ

- เป็นตัวเลือกหนึ่งที่จะช่วยในการนำเสนอ โดยจะมีโครงสร้างเดียวกันที่ได้มาตรฐาน และการจัดการข้อมูลรูปแบบคำสั่งซึ่งจะมีขอบเขต ได้แก่
 - หมวดหมู่ชี้แจง
 - ข้อกำหนดความต้องการที่เป็นไปตามรูปแบบ
 - เหตุผลของความต้องการที่ช่วยให้เข้าใจ และสามารถตรวจสอบได้

การเขียนโดยใช้ภาษาธรรมชาติที่เป็นมาตรฐานสากล

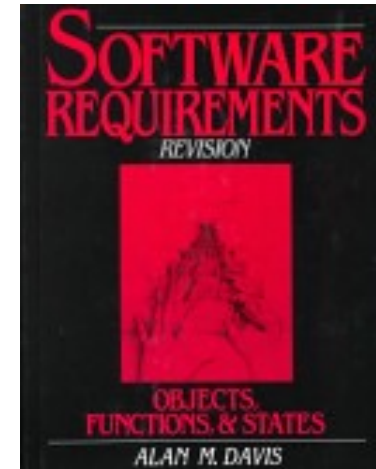
- นอกเหนือจากมาตรฐานเฉพาะที่ใช้ในการใช้ภาษาธรรมชาติที่กล่าวไว้ก่อนหน้านี้ อาจมีการใช้มาตรฐานสากลในการทำเอกสารความต้องการ (RD) ได้แก่
 - กฎของการรวมกลุ่มกัน (Grouping rules)
 - เทมเพลตเอกสาร RD แบบเป็นสากล (Global requirements document templates)

กฎของการรวมกลุ่มกัน

- เป็นการจัดกลุ่มของส่วนหัวข้อ หรือบท
 - มีความใกล้เคียงและเกี่ยวข้อง รวมไว้ในบทเดียวกัน
- เป็นมาตรฐานสากล เช่น มาตรฐานของ Davis ปี ค.ศ. 1993

ครอบคลุมเรื่องต่าง ๆ ได้แก่

 - วัตถุประสงค์ของระบบ (System objective)
 - ส่วนประกอบต่าง ๆ ของระบบ (System component)
 - งาน (Task)
 - ฟังก์ชันการทำงานของซอฟต์แวร์ (Software feature)



เทมเพลตเอกสาร RD แบบเป็นสากล

- โครงสร้างแบบมาตรฐานของเอกสารความต้องการ (Institute of Electrical and Electronic Engineers : IEEE, 1998) Std-830
- ส่วนสำคัญของ IEEE Std-830 ได้แก่
 1. บทนำ (Introduction)
 2. รายละเอียดทั่วไป (General description)
 3. ข้อกำหนดของความต้องการ (Specific requirements)
- ส่วนอื่น ๆ ของ IEEE Std-830 ได้แก่
 - ภาคผนวก (Appendices)
 - ดัชนี (Index)

1. บทนำ (Introduction)

- 1.1 วัตถุประสงค์ของเอกสารความต้องการ (Purpose of the requirements document)
- 1.2 ขอบเขตการดำเนินการ (Scope of the product)
- 1.3 นิยาม อักษรย่อ และคำย่อ (Definitions, acronyms and abbreviations)
- 1.4 การอ้างอิง (References)
- 1.5 การอธิบายภาพรวมของส่วนเพิ่มเติมในเอกสาร (Overview of the remainder of the document)

1.1 วัตถุประสงค์ของเอกสารความต้องการ

- บอกจุดมุ่งหมายของเอกสารความต้องการ
- บอกวัตถุประสงค์ต่าง ๆ ของผู้ใช้ว่าต้องการทำอะไรกับระบบ

1.2 ขอบเขตการดำเนินการ

- บอกขอบเขตของระบบ ได้แก่ ขอบเขตโดเมน และวัตถุประสงค์ของ System-to-be
- ทำให้ทราบถึงเงื่อนไข และภาพรวมการทำงานของระบบ

1.3 นิยาม อักษรย่อ และคำย่อ

- มีการอธิบายคำนิยามต่าง ๆ
- มีการอธิบายคำย่อ และขยายความให้เข้าใจยิ่งขึ้น
- มีการอธิบายตัวย่อของคำศัพท์ที่ใช้

1.4 การอ้างอิง

- มีการอ้างอิงและระบุแหล่งที่มา
- เพื่อให้ทราบถึงความเป็นมาทั้งหมดภายในระบบ

1.5 การอธิบายภาพรวมของส่วนเพิ่มเติมในเอกสาร

- มีการอธิบายภาพรวมของส่วนเพิ่มเติมในเอกสาร
- อาจมีหรือไม่มีส่วนเพิ่มเติมที่เพิ่มขึ้นมาในเอกสารก็ได้

2. รายละเอียดทั่วไป (General description)

- 2.1 ภาพรวมของระบบ (Product perspective)
- 2.2 การทำงานของระบบ (Product functions)
- 2.3 คุณลักษณะของผู้ใช้ระบบ (User characteristics)
- 2.4 เงื่อนไขทั่วไป (General constraints)
- 2.5 สมมติฐานและส่วนเพิ่มเติมของระบบ (Assumptions and dependencies)
- 2.6 ส่วนเพิ่มเติมของความต้องการ (Apportioning of requirements)

2.1 ภาพรวมของระบบ

- อธิบายภาพรวมของระบบ ได้แก่ ขอบเขตของซอฟต์แวร์ อินเทอร์เฟซของผู้ใช้ รวมถึงอุปกรณ์ และซอฟต์แวร์ที่ใช้

2.2 การทำงานของระบบ

- อธิบายฟังก์ชันการทำงานของระบบ
- ระบุฟังก์ชันที่ใช้ในการทำงาน ได้แก่ ฟังก์ชันการทำงานของ System-to-be

2.3 คุณลักษณะของผู้ใช้ระบบ

- อธิบายลักษณะของผู้ใช้ระบบ ในมิติที่เกี่ยวข้อง ได้แก่ สมมติฐานต่าง ๆ ที่เกี่ยวข้องกับผู้ใช้
- ระบุว่าผู้ใช้งานระบบ สามารถใช้งานในส่วนใดได้บ้างและใช้งานอย่างไร

2.4 เงื่อนไขทั่วไป

- ระบุข้อกำหนดที่ใช้ในการพัฒนา ได้แก่ เงื่อนไขของฮาร์ดแวร์ รวมถึง ระเบียบข้อบังคับ และระบบปฏิบัติการที่ใช้

2.5 สมมติฐานและส่วนเพิ่มเติมของระบบ

- อธิบายสมมติฐานและส่วนเพิ่มเติมของระบบ ที่จำเป็นต้องมี
- เช่น สมมติฐานสภาพแวดล้อม ซึ่งอาจเปลี่ยนแปลงได้ตลอดเวลา

2.6 ส่วนเพิ่มเติมของความต้องการ

- ระบุความต้องการเพิ่มเติม ได้แก่ ตัวเลือก หรือข้อผ่อนผันต่าง ๆ
- อาจมีหรือไม่มีส่วนเพิ่มเติมที่เพิ่มขึ้นมาในเอกสารก็ได้

3. ข้อกำหนดของความต้องการ (Specific requirements)

- 3.1 ความต้องการหลักของผู้ใช้ (Functional requirements)
- 3.2 ความต้องการด้านส่วนติดต่อกับภายนอก (External Interface requirements)
- 3.3 ความต้องการด้านประสิทธิภาพ (Performance requirements)
- 3.4 เงื่อนไขด้านการออกแบบ (Design constraints)
- 3.5 คุณลักษณะของซอฟต์แวร์ที่มีคุณภาพ (Software quality attributes)
- 3.6 ความต้องการอื่น ๆ (Other requirements)

3.1 ความต้องการหลักของผู้ใช้

- บอกความต้องการหลักของผู้ใช้
- มีการนำความต้องการของผู้ใช้มาออกแบบให้เป็นการทำงานหลักของระบบ

3.2 ความต้องการด้านส่วนติดต่อกับภายนอก

- บอกความต้องการด้านส่วนติดต่อกับภายนอก
- เป็นส่วนที่แสดงการติดต่อสื่อสารกับผู้ใช้

3.3 ความต้องการด้านประสิทธิภาพ

- บอกความต้องการด้านประสิทธิภาพ ได้แก่ ประสิทธิภาพด้านเวลา และการตอบสนอง
- แสดงให้เห็นถึงคุณภาพในการทำงานด้านต่าง ๆ ของระบบ

3.4 เงื่อนไขด้านการออกแบบ

- บอกเงื่อนไขด้านการออกแบบ ได้แก่ ข้อกำหนดความต้องการในการพัฒนา
- เพื่อให้ทราบขอบเขตของการออกแบบทั้งระบบ

3.5 คุณลักษณะของซอฟต์แวร์ที่มีคุณภาพ

- บอกคุณลักษณะของซอฟต์แวร์ที่มีคุณภาพ ได้แก่ ความต้องการทางด้านคุณภาพ
- เพื่อให้ทราบถึงคุณภาพของระบบด้านต่าง ๆ

3.6 ความต้องการอื่น ๆ

- ความต้องการอื่น ๆ ได้แก่ การรักษาความปลอดภัย ความน่าเชื่อถือ และการบำรุงรักษา

เทมเพลตมาตรฐานอื่น

- MIL-STD-498 (Military-Standard-498) ของกองทัพบกสหรัฐอเมริกา
- Nasa's SMAP-DID-P200-SW ขององค์การบริหารการบิน และอวกาศแห่งชาติ ประเทศสหรัฐอเมริกา หรือนาซา
- ESA's PSS-05 ของสถาบัน ESA (Software Engineering and Standardisation)

การใช้มาตรฐานสากลในการทำเอกสารความต้องการ

(Global rules on organization the requirement document)

- จะช่วยให้เข้าถึงปัญหาได้ง่ายขึ้น
- จะช่วยลดความกำกวมของเอกสาร
- มีเกณฑ์ในการวัดคุณภาพงานที่สูงขึ้น
- มีตัวอย่างโครงสร้าง RD เพื่อเป็นแนวทางในการจัดทำเอกสาร
- ทำให้มั่นใจว่าได้เอกสารที่มีมาตรฐาน

การใช้สัญลักษณ์แผนภาพ

(Use of diagrammatic notations)

การใช้สัญลักษณ์แผนภาพ

- เอกสารความต้องการยังคงเป็นภาษากึ่งทางการ
- แต่เดิมยังคงใช้แผนภาพ และความสัมพันธ์อย่างไม่เป็นทางการ โดยใช้เป็นรูปแบบเครื่องหมายสัญลักษณ์แบบกราฟิกทั่วไป ทำให้เกิดข้อผิดพลาดในการสื่อสาร
- UML หรือภาษากึ่งทางการสามารถใช้ได้ในเอกสารความต้องการ
- ในทางวิศวกรรมซอฟต์แวร์มีแผนภาพหลายแบบ ซึ่งขึ้นอยู่กับการใช้งานและความต้องการในการสื่อสาร
- เพื่อให้ผู้มีส่วนได้ส่วนเสียมีความเข้าใจตรงกัน

ตารางที่ 6-2 ชนิดของงานกับแผนภาพที่ใช้

ชนิด/ประเภท ของงาน	แผนภาพที่ใช้
ขอบเขตของระบบ (System scope)	Context diagrams, Problem diagrams
โครงสร้างของแนวคิด (Conceptual structures)	Entity relationship diagrams
กิจกรรมและข้อมูล (Activities and data)	Structured analysis and design technique diagram
การไหลของสารสนเทศ (Information flows)	Dataflow diagrams

ตารางที่ 6-2 ชนิดของงานกับแผนภาพที่ใช้ (2)

ชนิด/ประเภท ของงาน	แผนภาพที่ใช้
การทำงานของระบบ (System operations)	Use case diagrams
พฤติกรรมของระบบ (System behaviors)	State machine diagrams
สิ่งเร้าและการตอบสนอง (Stimuli and responses)	R-net diagrams

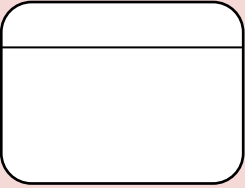


ขอบเขตของระบบ

- สามารถอธิบายขอบเขตของระบบได้โดยใช้แผนภาพต่อไปนี้
 - Context diagrams
 - Problem diagrams

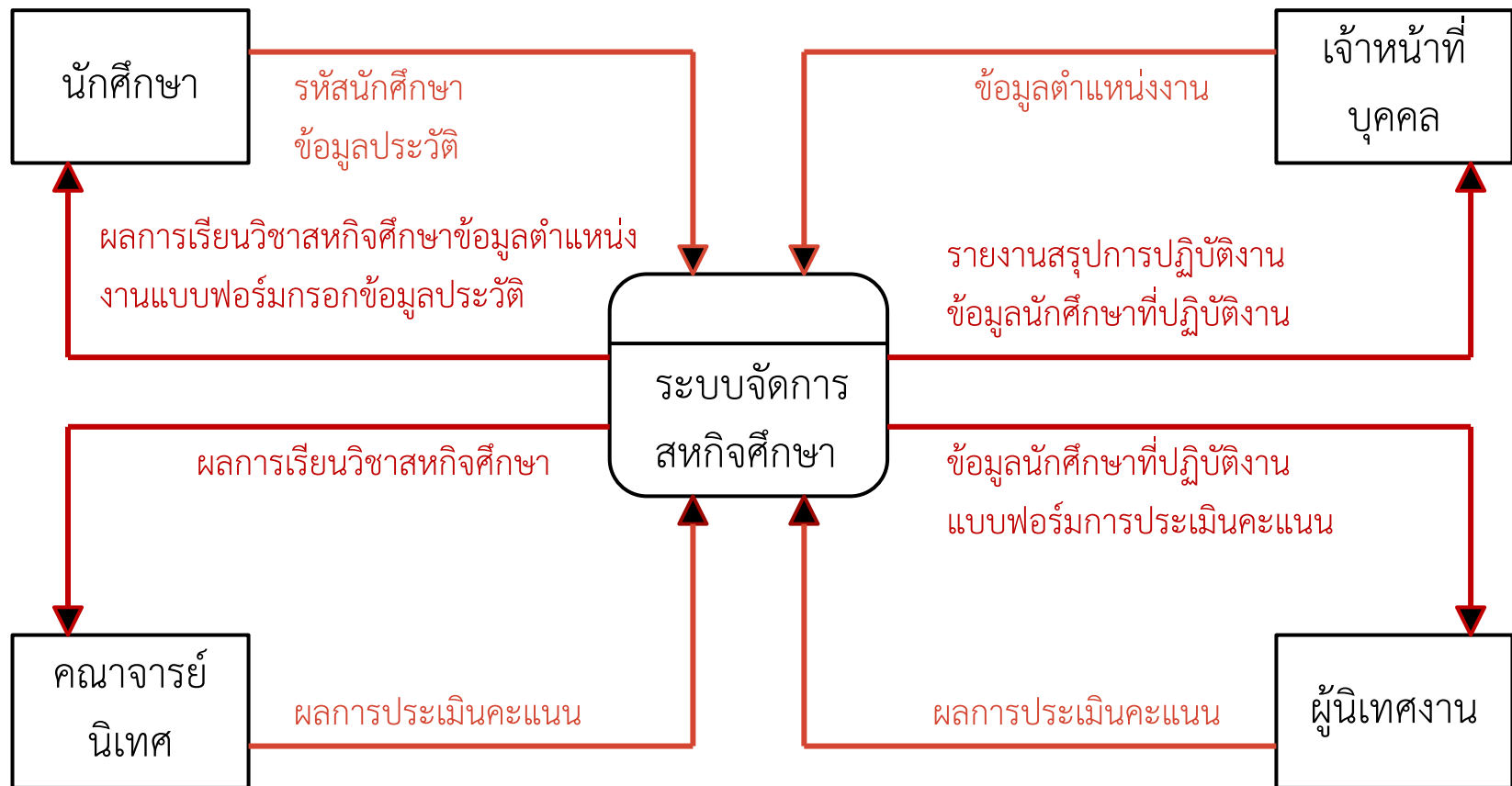
Context diagrams

- คือ แผนภาพการไหลของข้อมูลที่แสดงภาพรวมการทำงานของระบบที่มีความสัมพันธ์กับสภาพแวดล้อมภายนอก ระบบ ทั้งยังแสดงให้เห็นขอบเขตและเส้นแบ่งเขตของระบบที่กำลังพัฒนา

ตารางที่ 6-3 สัญลักษณ์ของ Context diagrams

สัญลักษณ์	ความหมาย
	กระบวนการ (Process) คือ ขั้นตอนการทำงานของระบบ
	สิ่งภายนอก (External entity) คือ สัญลักษณ์ของบุคคล หน่วยงานในองค์กร องค์กรอื่น ๆ หรือระบบงานอื่น ๆ ที่อยู่ภายนอกขอบเขตของระบบ แต่มีความสัมพันธ์กับระบบ
	ทิศทางของข้อมูล (Data flow) คือ สัญลักษณ์ที่ใช้อธิบายเส้นทางการไหลของข้อมูล มีเส้นตรงที่ประกอบด้วยหัวลูกศรตรงปลาย เพื่อบอกทิศทางการเดินทาง หรือการไหลของข้อมูล





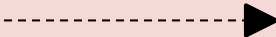
ภาพที่ 6-4 Context diagrams ระบบจัดการสหกิจศึกษา



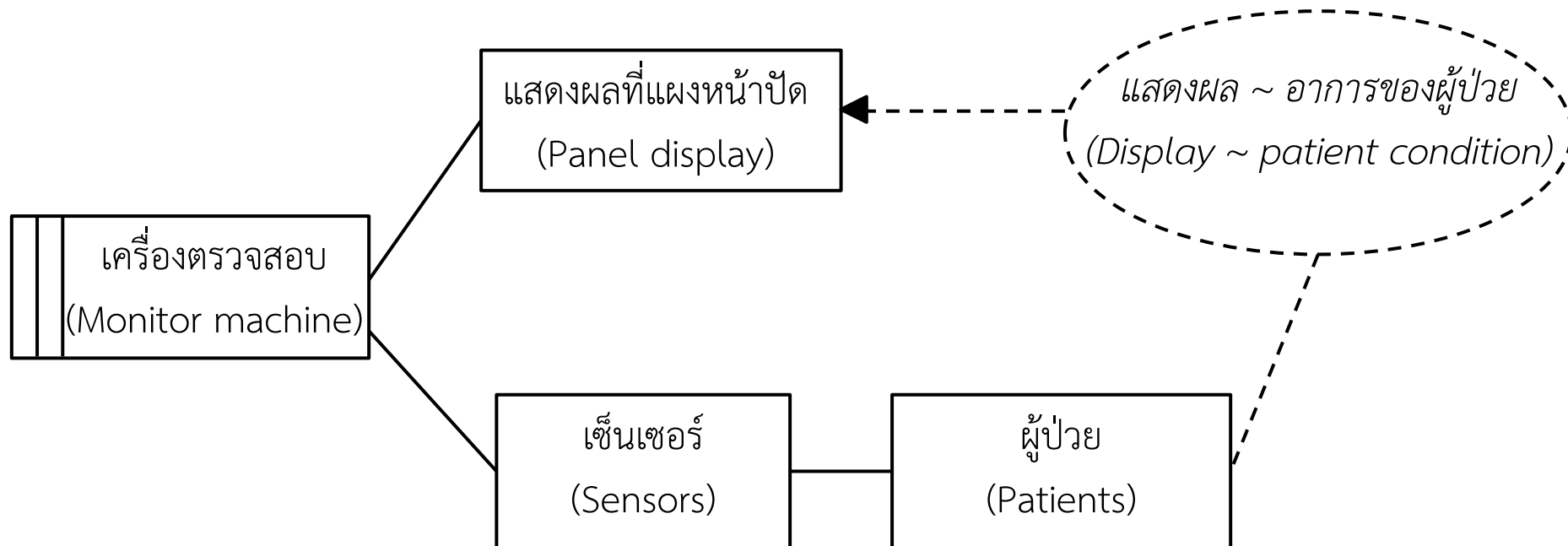
Problem diagrams

- เกิดจากส่วนควบคุมเหตุการณ์ที่ใช้ร่วมกันใน Context diagrams
- ถือว่าเป็นองค์ประกอบที่ได้รับผลกระทบจากความต้องการ และกลายเป็นปัญหาที่เกิดขึ้น
- แผนภาพที่เกิดขึ้นนั้นเรียกว่า “Problem diagrams” ถูกนำเสนอโดย Jackson ในปี ค.ศ. 2001

ตารางที่ 6-4 สัญลักษณ์ของ Problem diagrams

สัญลักษณ์	ความหมาย
	ตรวจสอบ (โดย Machine) คือ สัญลักษณ์ของการตรวจสอบข้อมูล
	ขอบเขตของปัญหา (Problem domain) คือ สัญลักษณ์ที่ระบุปัญหา
	ความต้องการ (Requirements) คือ สัญลักษณ์ของความต้องการที่เกี่ยวข้องกับผลกระทบบางอย่างในโดเมนของปัญหา
	ความสัมพันธ์ (Association) คือ สัญลักษณ์ที่แสดงความสัมพันธ์
	สัญลักษณ์การอ้างถึงความต้องการที่เชื่อมต่อกับโดเมนของปัญหาและเงื่อนไข

ภาพที่ 6-5 Problem diagrams ระบบคอมพิวเตอร์ ในโรงพยาบาล



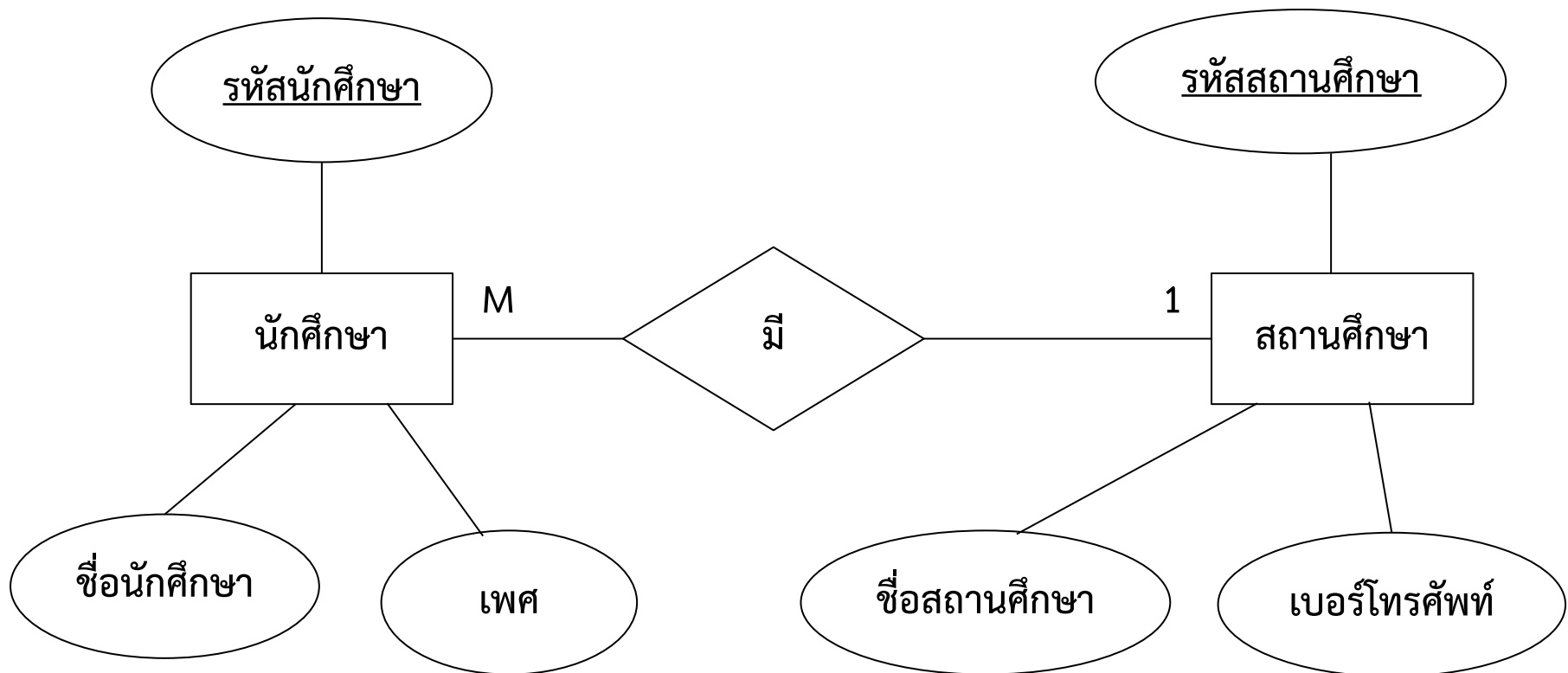
โครงสร้างของแนวคิด

- แบบจำลองความสัมพันธ์ของเอนทิตี (Entity relationship model) ถูกนำเสนอโดย Chen ในปี ค.ศ. 1976 และได้รับการพัฒนาต่อโดยอดจนสมบูรณ์ในปี ค.ศ. 1979
- ต่อมาได้พัฒนาโดยสร้างเป็นแผนภาพที่ใช้อธิบายโครงสร้างเชิงสัมพันธ์ที่รู้จักกันในชื่อ “Entity relationship diagrams” หรือ ER diagrams

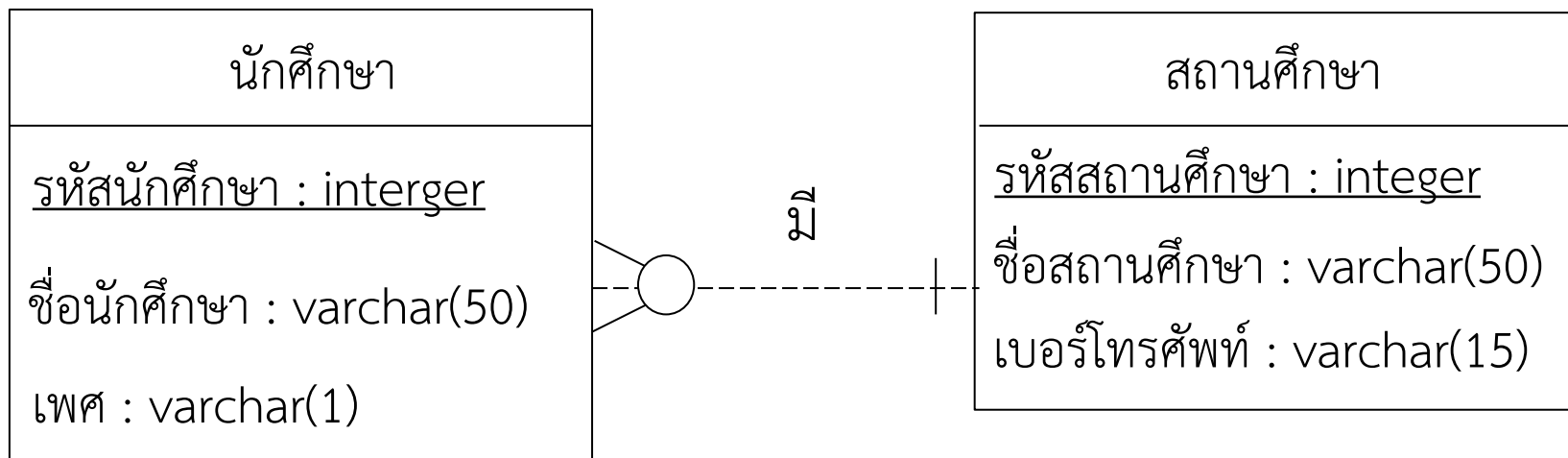
Entity relationship diagrams (ER)

- เอนทิตี (Entity) คือ ข้อมูลที่แสดงถึงคุณสมบัติของเอนทิตี
- แอตทริบิวต์ (Attributes) คือ คุณสมบัติของวัตถุ หรือสิ่งของ โดยอธิบายรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับลักษณะของเอนทิตี โดยคุณสมบัตินี้มีอยู่ในทุกเอนทิตี
- ความสัมพันธ์ (Relationship) คือ เอนทิตีที่จะต้องมีความสัมพันธ์ร่วมกัน โดยจะมีชื่อแสดงความสัมพันธ์

ภาพที่ 6-7 ER diagrams ระหว่างนักศึกษา กับสถาน ประกอบการของรูปแบบ Chen model



ภาพที่ 6-8 ER diagrams ระหว่างนักศึกษา กับสถานศึกษา รูปแบบ Crow's foot model




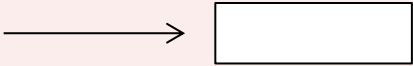

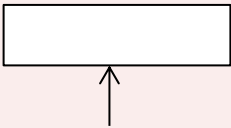
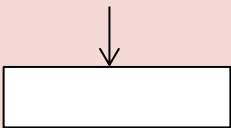
กิจกรรมและข้อมูล

- Structured analysis and design technique diagram หรือ SADT diagram
- เป็นแผนภาพกิจกรรมที่ได้จากการประมวล การวิเคราะห์โครงสร้าง และเทคนิคการออกแบบ
- แบ่งออกเป็น 2 ชนิด ได้แก่
 - Actigrams
 - Datagrams

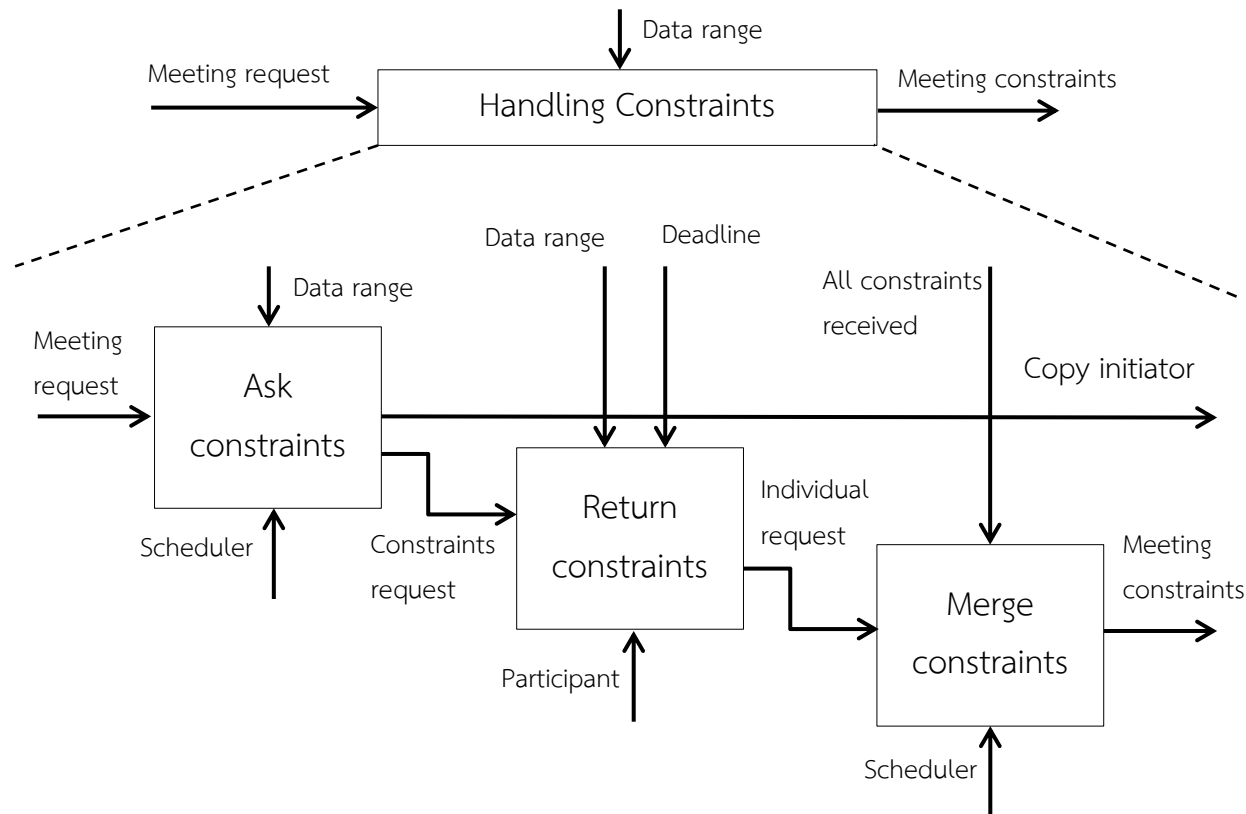
Actigram

- สามารถระบุกิจกรรมของระบบที่สามารถแบ่งออกเป็นกิจกรรมย่อย ๆ ได้
- เป็นการประกาศกิจกรรมโดยมีข้อมูลนำเข้า/ส่งออก และมีการเชื่อมต่อผ่านการเชื่อมโยงข้อมูล

ตารางที่ 6-5 สัญลักษณ์ของ Actigram

สัญลักษณ์	ความหมาย
	กิจกรรม (Activity)
	ข้อมูลนำเข้า (Input data)
	ข้อมูลส่งออก (Output data)
	ตัวดำเนินการ (Processor)
	ตัวควบคุมข้อมูล (Controlling data)

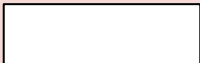


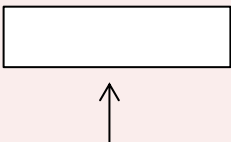
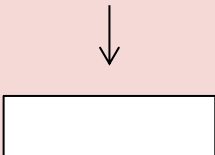
ภาพที่ 6-9 Actigram ของ Handling Constraints



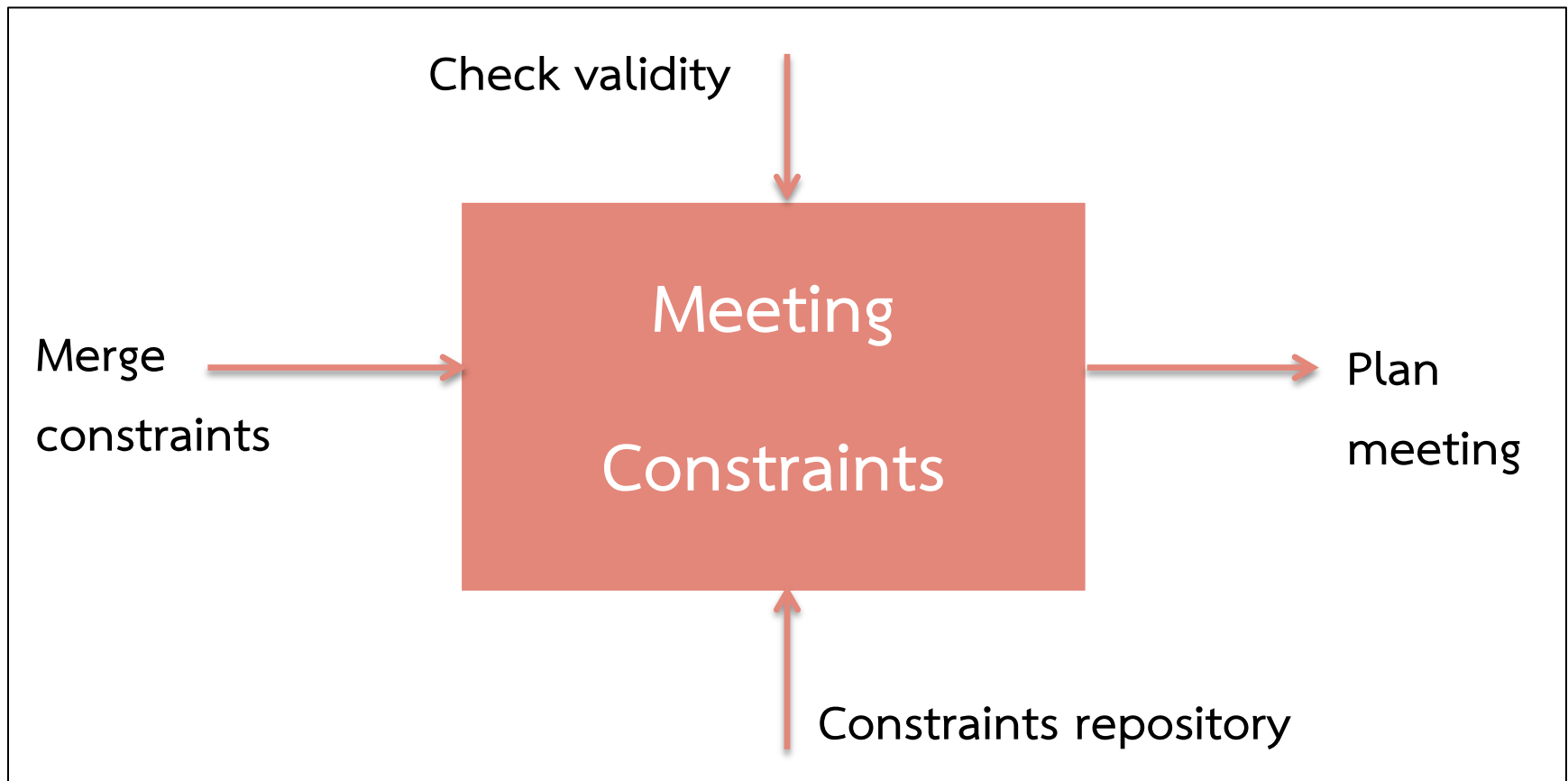
Datagram

- เป็นการประกาศข้อมูลในระบบจากการพัฒนาซอฟต์แวร์ และมีการเชื่อมต่อกันผ่านตัวควบคุมการเชื่อมโยง
- จะสามารถตรวจสอบความสอดคล้อง และครบถ้วนของข้อมูลได้จากข้อมูล ได้แก่
 - ข้อมูลนำเข้า/ส่งออกของกิจกรรมหลัก จะต้องปรากฏในข้อมูลนำเข้า/ส่งออกของกิจกรรมย่อย เพื่อให้มีความสอดคล้องกัน
 - กิจกรรมใด ๆ จะต้องมีการป้อนข้อมูล และการมีการส่งออก
 - กิจกรรมการควบคุมใน Datagram จะต้องกำหนดไว้ใน Actigram

ตารางที่ 6-6 สัญลักษณ์ของ Datagram

สัญลักษณ์	ความหมาย
	ข้อมูล (Data)
	กิจกรรมที่นำมาใช้กับข้อมูล (Producing activity)
	ผลลัพธ์ของกิจกรรม (Consuming activity)
	ทรัพยากร (Resource)
	ตัวควบคุมกิจกรรม (Controlling activity)

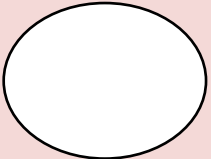



ภาพที่ 6-10 Datagram ของ Meeting Constraints



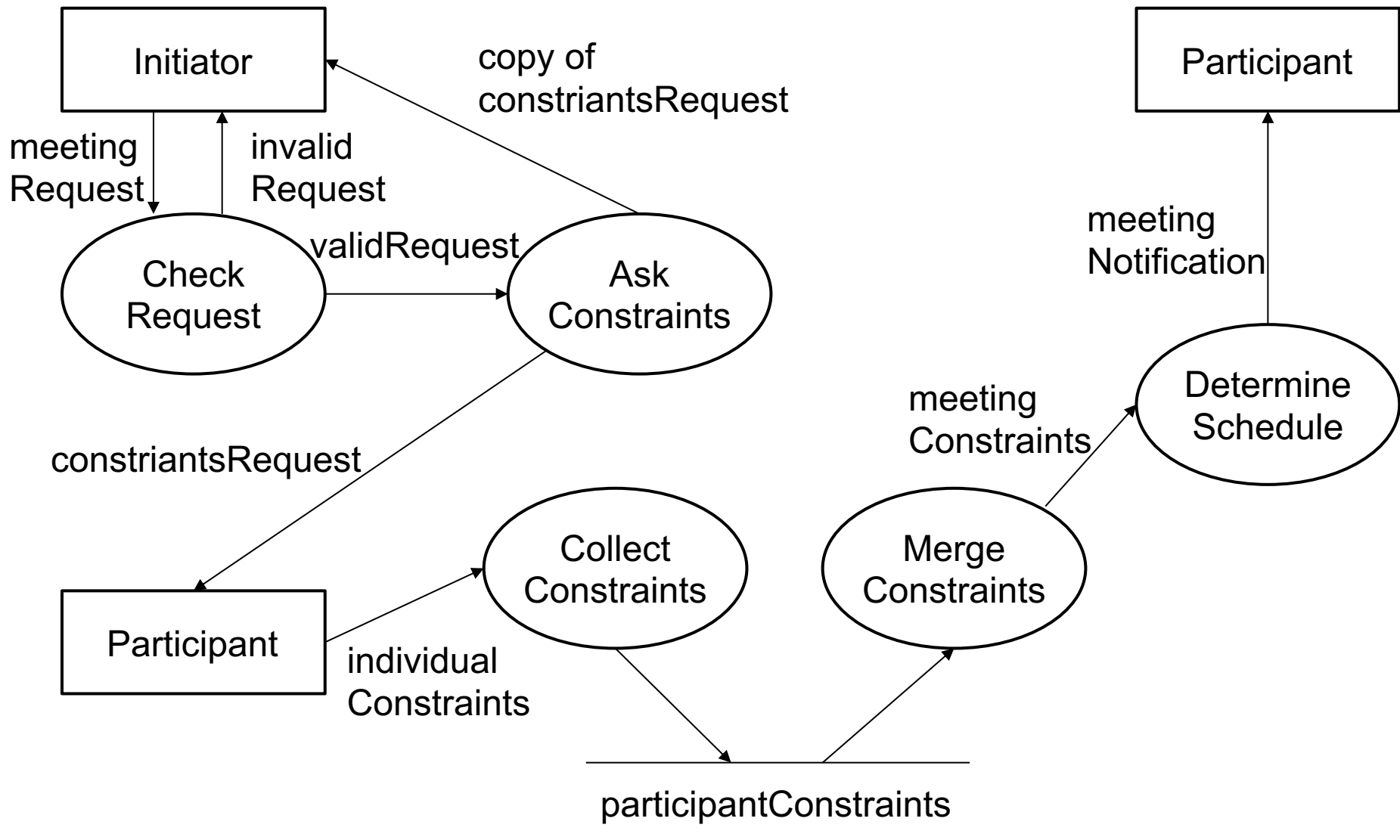
แผนภาพการไหลของข้อมูล (Dataflow Diagram : DFD)

- DFD เป็นเครื่องมือที่ใช้เพื่อแสดงการไหลของข้อมูล และการประมวลผลต่าง ๆ ในระบบ
- โดย DFD ต้องสัมพันธ์กับแหล่งเก็บข้อมูลที่ใช้
- เป็นสื่อที่ช่วยให้การวิเคราะห์ระบบเป็นไปได้อย่างง่าย
- ช่วยให้ความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบ โปรแกรมเมอร์ และผู้ใช้ระบบ

ตารางที่ 6-7 สัญลักษณ์ของ Dataflow diagram

สัญลักษณ์	ความหมาย
	การประมวลผล (Process)
	กระแสนำเข้า/ส่งออกของข้อมูล (Data flow)
	ที่เก็บข้อมูล (Data Store)
	สิ่งที่อยู่ภายนอกระบบ (External entity)

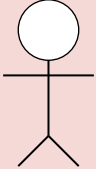
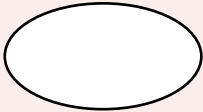

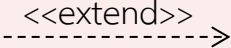
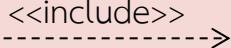
ภาพที่ 6-11 Dataflow diagram ของระบบจัดการประชุม



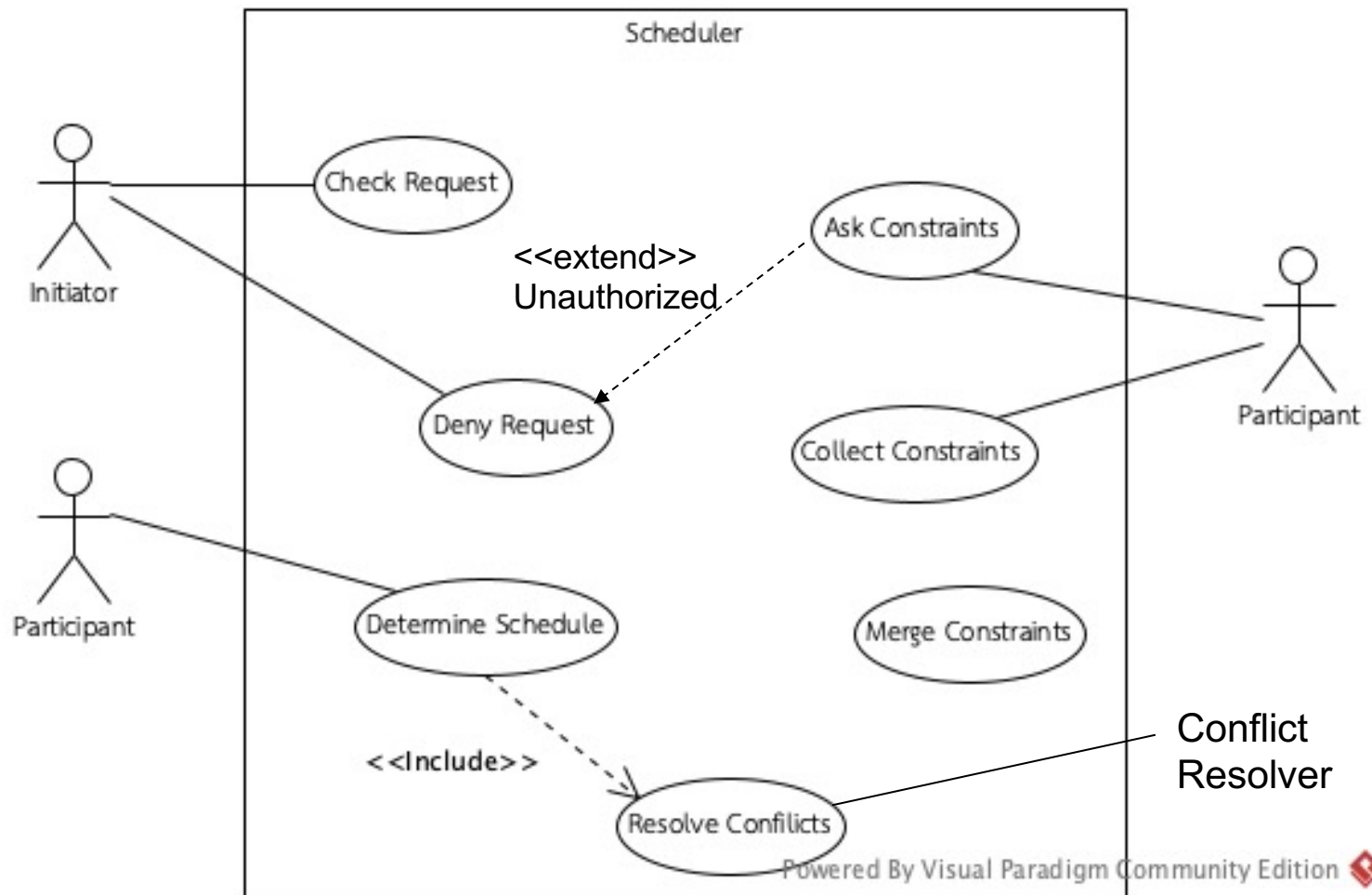
Use case diagram

- คือ แผนภาพที่แสดงการทำงานของผู้ใช้ระบบ (User) กับความสัมพันธ์ของระบบย่อย (Sub systems) ภายในระบบใหญ่
- จุดประสงค์หลักของการเขียน Use case diagram ได้แก่
 - เพื่อเล่าเรื่องราวทั้งหมดของระบบว่ามีการทำงานอย่างไรบ้าง
 - เป็นการนำ Requirements ของผู้ใช้ หรือสถานการณ์ต่าง ๆ ของระบบมาออกแบบ
 - ถือว่าเป็นจุดเริ่มต้นในการวิเคราะห์ และการออกแบบระบบ

ตารางที่ 6-8 สัญลักษณ์ของ Use case diagram

สัญลักษณ์	ความหมาย
	Actor ผู้ใช้ระบบ (User)
	Use case ระบบการทำงานย่อย (Sub system)
	เชื่อม Actor กับ Use case เพื่อแสดงการใช้งานของ Use case และ Actor
	เชื่อมต่อระหว่าง Use case หนึ่งที่มีผลกับการทำงานปกติของอีก Use case หนึ่ง เช่น ถ้าทำ Use case A อาจจะทำ Use case B ด้วยหรือไม่ก็ได้
	เชื่อมต่อระหว่าง Use case ที่มีการทำงานร่วมกัน เช่น ถ้าทำ Use case A จะต้องทำ Use case B ด้วยเสมอ

ภาพที่ 6-12 Use-case diagram การจัดการประชุม







State machine diagram

- ใช้เพื่อแสดงสถานะของวัตถุรวมไปถึงเหตุการณ์ต่าง ๆ ที่สามารถทำให้สถานะของวัตถุนั้นเปลี่ยนแปลงไป และการกระทำที่เกิดขึ้นเมื่อสถานะของระบบมีการเปลี่ยนแปลง
- ใช้เพื่อแสดงกิจกรรมในฟังก์ชันต่าง ๆ ของคลาส ได้แก่
 - แสดงพฤติกรรมของวัตถุที่สร้างขึ้นจากคลาส
 - แสดงสถานะ (State) ทั้งหมดที่เป็นไปได้ของวัตถุ
 - แสดงการเปลี่ยนแปลงสถานะของวัตถุ

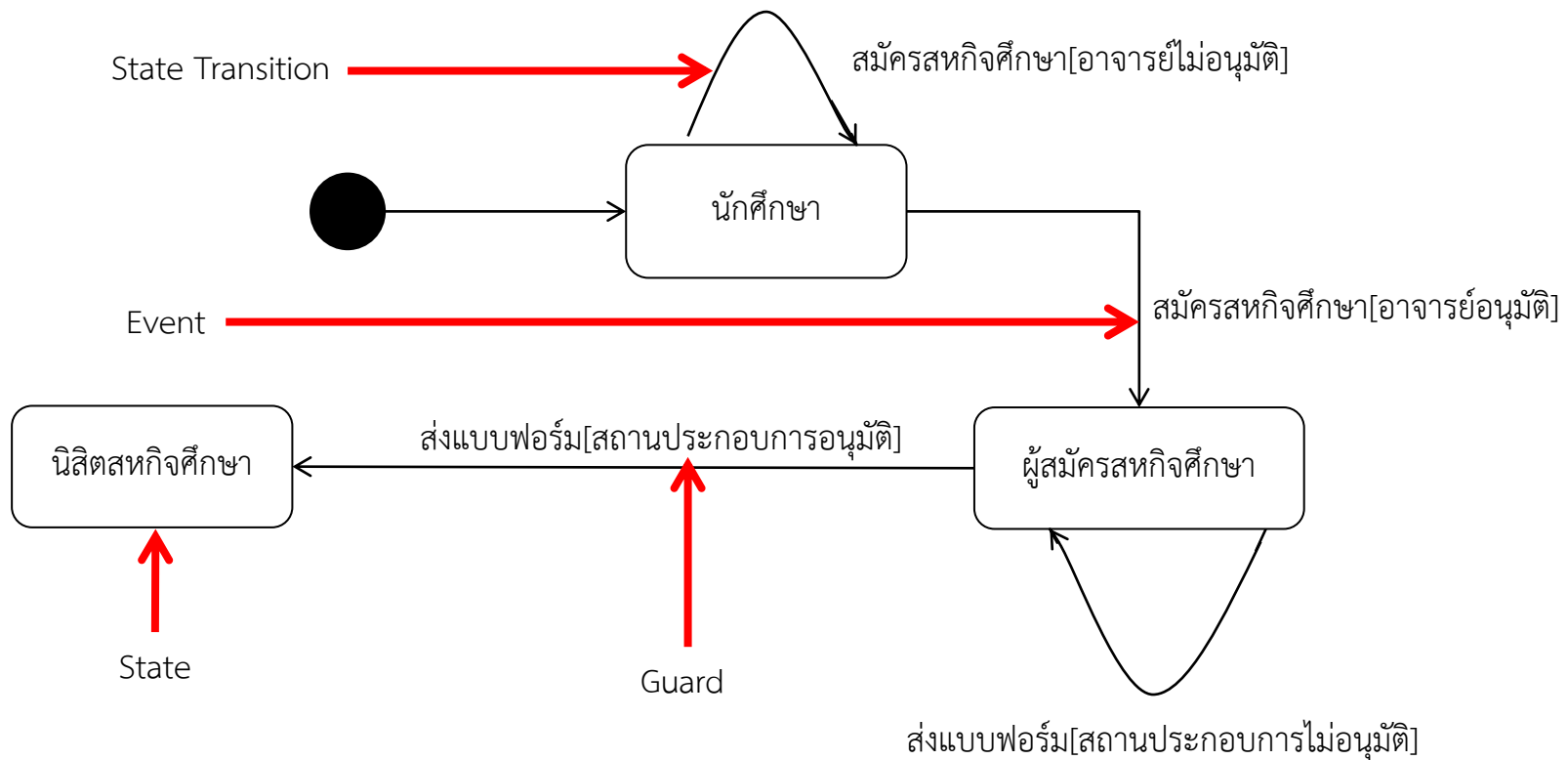
State machine diagram (2)

- การที่ Object เปลี่ยนจากสถานะหนึ่งไปยังอีกสถานะหนึ่ง จะทำให้เกิดกิจกรรมขึ้นใน Object นั้น
- กิจกรรมโดยรวมของระบบเกิดจากกิจกรรมย่อยของ Object แต่ละตัวรวมกัน
- กิจกรรมที่เกิดขึ้นใน Object เกิดจาก 2 สิ่งประกอบกัน ได้แก่
 - สถานะ (State)
 - การเปลี่ยนสถานะ (Transitions)

ตารางที่ 6-10 สัญลักษณ์ของ State diagram

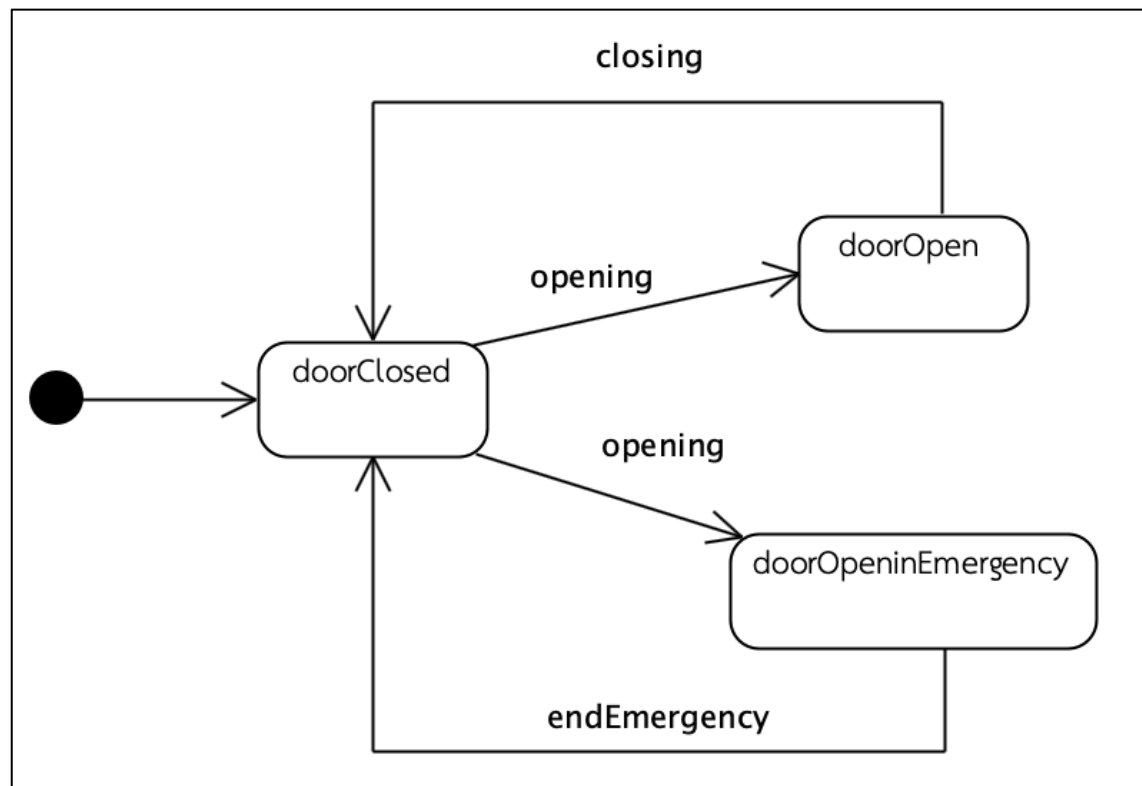
สัญลักษณ์	ความหมาย
	สถานะ (State)
	จุดเริ่มต้นของสถานะ (Initial state) คือ สถานะเริ่มต้นที่ Object
	จุดสิ้นสุดของการเปลี่ยนสถานะ (Terminal state) คือ สถานะที่ Object สิ้นสุด หรือจบการทำงาน
	เส้นแสดงการเปลี่ยนสถานะโดยระบุ Event หรือ Action ที่ทำให้เกิดการเปลี่ยนสถานะ

ภาพที่ 6-14 State diagram การจัดการสหกิจศึกษา



ภาพที่ 6-15 State machine diagram การเปิด-ปิดประตู

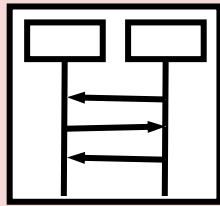
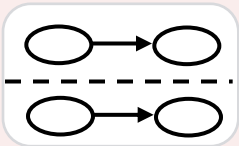
Statechart ที่ไม่สามารถตัดสินใจได้ ลักษณะนี้เรียกว่า **Non-deterministic**



ตารางที่ 6-12 แผนภาพในกระบวนการ RE ตามมาตรฐานของ UML

แผนภาพในกระบวนการ RE	แผนภาพตามมาตรฐาน UML	ความหมาย	แผนภาพ
ER diagram	Class diagram	ใช้เพื่อแสดง Class และความสัมพันธ์ในแง่มุมต่าง ๆ ระหว่าง Class	
Use case diagram	Use case diagram	ใช้เพื่อแสดงให้เห็นภาพรวมของระบบเป็นการตั้ง Requirements ของผู้ใช้เพื่อนำมาวิเคราะห์ออกแบบ	

ตารางที่ 6-12 แผนภาพในกระบวนการ RE ตามมาตรฐานของ UML (2)

แผนภาพในกระบวนการ RE	แผนภาพตามมาตรฐาน UML	ความหมาย	แผนภาพ
ET diagram	Sequence diagram	ใช้เพื่อแสดงให้เห็นถึงการปฏิสัมพันธ์ (Interaction) ระหว่าง Objects ต่างๆ ของคลาส (Sequence) ที่เกิดเหตุการณ์ขึ้นตามลำดับของเวลา	
SM diagram	State diagram	ใช้เพื่อแสดงสถานะของวัตถุรวมถึงเหตุการณ์ต่าง ๆ ที่มีการเปลี่ยนแปลงสถานะของการทำงาน	

ข้อดีและเงื่อนไขของการใช้แผนภาพ

- ข้อดี
 - เพื่อช่วยให้เข้าใจง่ายขึ้น
 - เป็นภาพรวมและโครงสร้างที่สำคัญ (Functional and structural aspects)
 - ช่วยระบุฟังก์ชันการทำงานและโครงสร้าง
- ข้อจำกัด
 - ภาษาอาจมีกำกวมซึ่งทำให้มีผลการวิเคราะห์ที่แตกต่างกัน
 - เป็นรูปแบบ (Forms) ที่จำกัดในการวิเคราะห์
 - มีข้อจำกัดด้านฟังก์ชันการทำงานหลักและโครงสร้าง (Functional and structural)

การจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ (Formal specification)



การจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ

- ประโยชน์ของการจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ ได้แก่
 - มีความแม่นยำ
 - สามารถให้ความหมายที่ชัดเจนมากขึ้น
 - มีเครื่องมือที่ใช้ในการตรวจสอบ และการทวนสอบ
- ในการเก็บรวบรวมความต้องการบางครั้งจะต้องระบุความต้องการให้เป็นระเบียบแบบแผน โดยทั่วไปจะจัดอยู่ในรูปแบบของบทที่เป็นหมวดหมู่ และมีความสัมพันธ์กัน

ข้อดีและข้อจำกัดของการจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ

- ข้อดี
 - เป็นวิธีการกำหนดอย่างเป็นทางการที่ทุกคนสามารถเข้าใจตรงกัน
 - มีการเขียนที่แม่นยำสำหรับการตีความ ส่งผลให้เห็นถึงความต้องการ การสมมติฐาน และการนิยามคุณสมบัติโดเมนของงานในรูปแบบเดียวกัน
- ข้อจำกัด
 - มีการใช้ภาษาที่เข้าใจได้ยาก
 - มีรายละเอียดยืดยาวค่อนข้างมาก โดยผู้อ่านจำเป็นต้องมีความรู้ความเชี่ยวชาญ
 - มีวิธีการกำหนดอย่างเป็นทางการจึงทำให้ผู้ใช้ไม่สามารถทำความเข้าใจได้

วิธีการจัดทำข้อกำหนดในรูปแบบที่เป็นทางการ

1. Logic as a basis for formalizing statements
2. History-based specification
3. State-based specification

1. Logic as a basis for formalizing statements

- การเขียนความต้องการในรูปแบบทางการโดยใช้วิธีการทางตรรกศาสตร์
- ตรรกะถูกสร้างขึ้นจาก 3 องค์ประกอบ ได้แก่
 - วากยสัมพันธ์ (Syntax) คือ กฎต่าง ๆ ที่ถูกกำหนดไว้ เพื่อให้ความต้องการเป็นระเบียบแบบแผนที่เหมือนกัน
 - ความหมาย (Semantics) คือ กฎต่าง ๆ ที่ระบุไว้ เพื่อให้ความต้องการมีความถูกต้อง
 - การใช้ทฤษฎีในการพิสูจน์ (Proof theory) ประกอบด้วย ชุดของกฎการอนุมานที่ช่วยให้เกิดข้อความใหม่ ๆ ที่จะได้รับจากผู้มีส่วนได้ส่วนเสีย

1. Logic as a basis for formalizing statements (2)

- ตรรกะเชิงประพจน์ช่วยให้สามารถเขียนข้อความหลาย ๆ ข้อความที่มีความสัมพันธ์ต่อกันผ่านด้วย**ตัวเชื่อม** (Connective) ได้แก่
 - ▣ \wedge (และ) \vee (หรือ) \sim (นิเสธ) \longrightarrow (ถ้า...แล้ว) และ \longleftrightarrow (ก็ต่อเมื่อ)
- ตัวเชื่อมสามารถใช้เชื่อมได้หลายประพจน์

2. History-based specification

- การจัดทำข้อกำหนดโดยใช้ข้อมูลในอดีต
- การอนุมานว่าประวัติโดยรวมมีความถูกต้อง และถูกยอมรับ

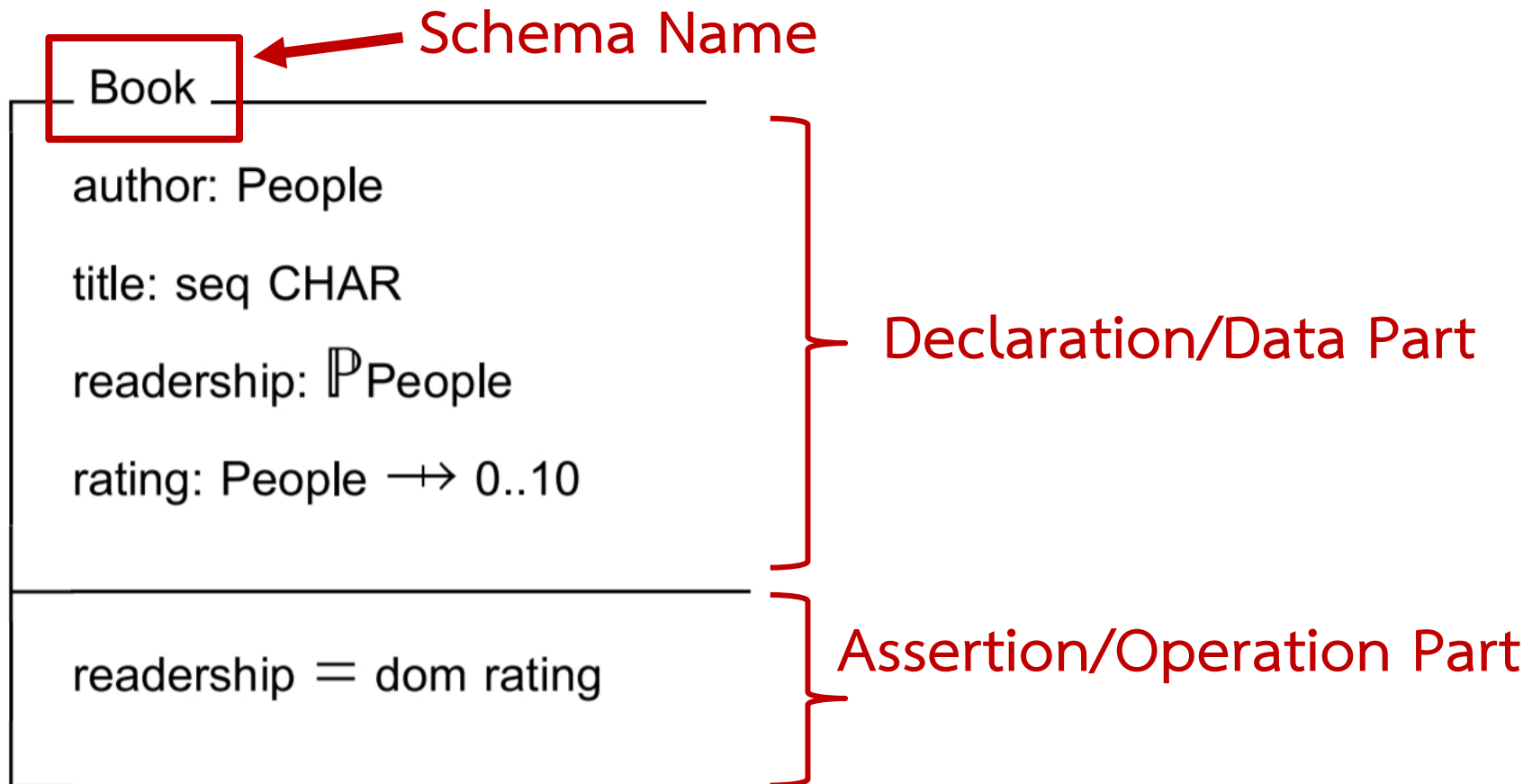
3. State-based specification

- การจัดทำข้อกำหนดโดยใช้สถานะของวัตถุ
- ข้อกำหนดของระบบที่ภาพรวมยอมรับจากความต้องการ สมมติฐาน และขอบเขตคุณสมบัติที่ระบุไว้ในตรรกะ
- โดยทั่วไปจะอธิบายในรูปแบบของภาษา Z ซึ่งจะเกี่ยวข้องกับ Schemas โดยมี 2 ชนิด ได้แก่
 - Data schemas
 - Operation schemas

ภาษา Z ซึ่งจะเกี่ยวข้องกับ Schemas

1. Data schemas เป็นโครงสร้างสำหรับการจัดระเบียบ และการจัดข้อมูล
ในฐานข้อมูล มีการกำหนดทั้งเนื้อหาข้อมูล และความสัมพันธ์
2. Operation schemas เป็นการระบุการดำเนินงาน หรือการทำงานของ
ระบบด้วยการประกาศตัวแปรอย่างมีลำดับ

ภาพที่ 6-18 โครงสร้างของ Schema



Declaration Part

- author : People คือ author เป็นหนึ่งในเซตของ People
- title : seq CHAR คือ เซตของชุดลำดับ (Sequence) ของ CHAR = { A,B,C, ...} ซึ่งอาจจะเป็นเซตต่อไปนี้เป็น {<C,O,M,P,U,T,E,R>, <S,O,F,T,W,A,R,E>} และตัวแปร title อาจจะมีค่าเป็น <C,O,M,P,U,T,E,R> เป็นต้น

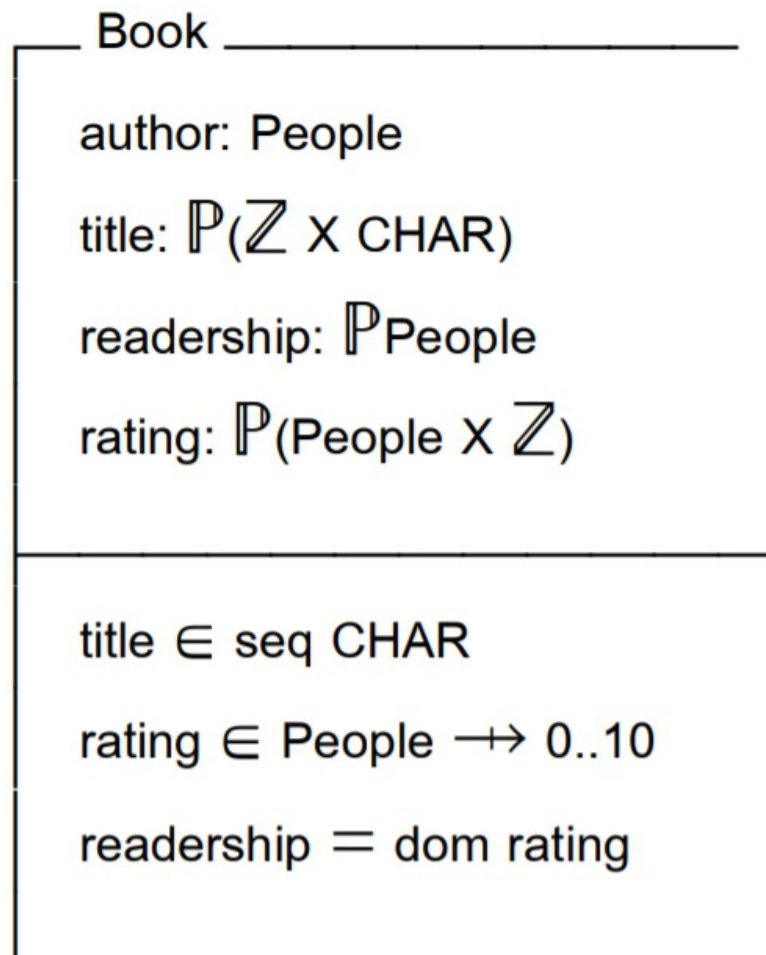
Declaration Part (2)

- readership : $\mathbb{P}\text{People}$ คือ พาวเวอร์เซตของ People
- rating : $\text{People} \rightarrow 0..10$ จะเป็นตัวแปรชนิดที่ระบุด้วยฟังก์ชันบางส่วน (Partial Function) ถ้ากำหนดให้ $\text{People} = \{\text{Popeye}, \text{Pluem}, \text{Pae}...\}$ แล้ว $\text{People} \rightarrow 0..10$ จะได้เซตดังนี้
- $\{ (\text{Popeye}, 1), (\text{Pluem}, 2), (\text{Pae}, 3), \dots \}$

Assertion Part

- readership = **dom** rating คือ การนำโดเมนในคู่อันดับของเซตใน rating ไปเพิ่มในเซต readership

ภาพที่ 6-19 โครงสร้างของ Z Schema



ตัวอย่างรูปแบบ และเทคนิคการเขียนเอกสาร

- การจัดทำสารบัญ
- การกรีนนำเข้าสู่เนื้อหา
- เครื่องมือที่ใช้ในการพัฒนา
- ขอบเขตของซอฟต์แวร์
- แผนการดำเนินงาน
- ดัชนีชี้วัดความสำเร็จ
- รายละเอียดของการปฏิบัติงาน
- แผนภาพต่าง ๆ

ตัวอย่างรูปแบบ และเทคนิคการเขียนเอกสาร (2)

- บรรณานุกรม
- ระดับชั้นที่ใช้การเขียนหัวข้อย่อย
- ระยะห่างในการย่อหน้า และเว้นวรรค
- การเขียนภาษาอังกฤษทุกคำขึ้นต้นด้วยตัวอักษรตัวใหญ่
- การใช้เครื่องหมาย , สำหรับการเขียนประโยคภาษาอังกฤษ
- การใช้เครื่องหมายไม้ยมก (๓)
- การใช้เครื่องหมายไปยาลน้อย (๓) และไปยาลใหญ่ (๓๓๓)
- การใช้เครื่องหมายวงเล็บ ()

ตัวอย่างรูปแบบ และเทคนิคการเขียนเอกสาร (3)

- คำสรรพนามในประโยค
- การใช้คำว่า “และ” สำหรับการเขียนประโยคภาษาไทย
- ชื่อรูปภาพ
- ชื่อตาราง
- การอ้างอิงรูปภาพ หรือตาราง
- ตัวอย่างตาราง และคำอธิบายตาราง

การจัดทำสารบัญ

- สารบัญเนื้อหา
- สารบัญรูปภาพ
- สารบัญตาราง

ภาพที่ 6-20 สารบัญ

สารบัญ	
	หน้า
กิตติกรรมประกาศ.....	ก
บทคัดย่อ	ข
สารบัญ	ค
สารบัญรูปภาพ	จ
สารบัญตาราง	ฉ
บทที่	
1 บทนำ	
1.1 ข้อมูลของบริษัท เอ็ม เอฟ อี ซี จำกัด (มหาชน).....	2
1.2 ปัญหาหรือความจำเป็นในการปฏิบัติงานสหกิจศึกษา.....	6
1.3 วัตถุประสงค์ของโครงการสหกิจศึกษาที่ได้รับมอบหมาย.....	7
1.4 เครื่องมือที่ใช้ในการพัฒนา.....	8

ภาพที่ 6-21 สารบัญรูปภาพ

สารบัญรูปภาพ	
ภาพที่	หน้า
1-1 ตราสัญลักษณ์ของบริษัท เอ็ม เอฟ อี ซี จำกัด (มหาชน).....	2
1-2 แผนที่ตั้งองค์กร	3
1-3 โครงสร้างการบริหารของบริษัท เอ็ม เอฟ อี ซี จำกัด (มหาชน)	4
2-1 ตัวอย่างการทำงานของ Reactive programming	19
2-2 การทำงานของ Observable.....	20
2-3 การทำงานของ MVVM pattern	23
2-4 Clean architecture.....	23

ภาพที่ 6-22 สารบัญตาราง

สารบัญตาราง

ตารางที่	หน้า
1-1 แผนปฏิบัติงานสหกิจศึกษา	14
2-1 คำศัพท์เฉพาะ	16
3-1 แผนการทดสอบ (Test Plan)	53
3-2 กรณียกสอบ ส่วนของการนำเข้าข้อมูลงานป้องกัน	55
3-3 กรณียกสอบ ส่วนของเปลี่ยนสถานะงานป้องกัน	56
3-4 กรณียกสอบ ส่วนของข้อมูลหลัก.....	57
3-5 กรณียกสอบ ส่วนของข้อมูลผู้เข้าซื้อและผู้ค้า	58

ภาพที่ 6-23 การกรีนนำเข้าสู่เนื้อหา

1.4. เครื่องมือและเทคโนโลยีที่ใช้ในการปฏิบัติงาน

การพัฒนาแอปพลิเคชันแบบ Web Based Application ได้รับความนิยมเป็นอย่างยิ่ง โดยที่ภาษาที่เกี่ยวข้องกับการพัฒนาเว็บถูกพัฒนามามากมายหลายภาษา ซึ่งแต่ละภาษาล้วนต่างมีข้อดีข้อเสียแตกต่างกันไป เพื่อรองรับความก้าวหน้าทางด้านเทคโนโลยีเว็บในยุคปัจจุบัน ซึ่งต่อไปนี้จะขอกล่าวรายละเอียดของเครื่องมือที่สามารถจำแนกได้ดังนี้

เครื่องมือที่ใช้ในการพัฒนา

- ภาษาที่ใช้ในการพัฒนา
- ซอฟต์แวร์ที่ใช้ในการพัฒนา
- เครื่องแม่ข่ายที่ใช้ในการพัฒนา

ภาพที่ 6-24 ภาษาที่ใช้ในการพัฒนา

1.4.1 ภาษาที่ใช้ในการพัฒนา

1) HTML (Hypertext Markup Language) ภาษาที่ใช้ในการสร้างเอกสารในภาพแบบของเว็บเพจบนระบบเครือข่ายอินเทอร์เน็ต โดยพัฒนาจากภาษา SGML (Standard Generalized Markup Language) ซึ่งการเขียน HTML นั้นจะมีโครงสร้างการเขียนโดยใช้ Tag ควบคุมการแสดงผลในภาพแบบต่างๆ เช่น ข้อความ ภาพ หรือวัตถุอื่นๆ เป็นต้น

2) JavaScript ภาษาที่ทำหน้าที่แปลความหมายและดำเนินการทีละคำสั่ง ภาษานี้มีชื่อเดิมว่า Live Script ถูกพัฒนาโดย Netscape Navigator ช่วยให้เว็บเพจสามารถแสดงเนื้อหา ที่มีการเปลี่ยนแปลงได้ตามเงื่อนไข หรือสภาพแวดล้อมที่แตกต่างกัน หรือโต้ตอบกับผู้ใช้ได้มากขึ้น

3) CSS (Cascading Style Sheet) ภาษาที่ใช้ในการจัดรูปแบบการแสดงผลของภาษา HTML เช่น สีของข้อความ สีของพื้นหลัง ขนาดตัวอักษร หรือประเภทตัวอักษร เป็นต้น

ภาพที่ 6-25 ซอฟต์แวร์ที่ใช้ในการพัฒนา

1.4.2 ซอฟต์แวร์ที่ใช้ในการพัฒนา

1) โปรแกรม Notepad++ เป็นโปรแกรมที่ช่วยในการพัฒนาระบบ มีลักษณะคล้ายกับโปรแกรม Notepad แต่เพิ่มความสามารถในการแก้ไข จัดการไฟล์ โดยมีภาพแบบหรือสีของตัวอักษรที่ง่ายต่อการพัฒนา และช่วยในการเดาคำศัพท์ เพื่ออำนวยความสะดวกในการพัฒนาระบบ

2) โปรแกรม Win SCP (Windows Secure Copy) เป็นซอฟต์แวร์สำหรับการจัดการไฟล์ระหว่างเครื่องแม่ข่ายกับเครื่องลูกข่าย ใช้ในการแก้ไข บันทึก หรือลบไฟล์เครื่องแม่ข่าย ทำให้เพิ่มความสะดวกในการจัดการไฟล์

3) โปรแกรม Putty เป็นซอฟต์แวร์เข้าใช้งานเครื่องแม่ข่าย (Server) ในระยะไกลในภาพแบบบรรทัดคำสั่ง (Command Line) ใช้ในการใช้คำสั่งสำหรับสำรองฐานข้อมูล และนำเข้าฐานข้อมูล เป็นต้น

ภาพที่ 6-26 เครื่องแม่ข่ายที่ใช้ในการพัฒนา

1.4.3 เครื่องแม่ข่ายที่ใช้ในการพัฒนา

ลักษณะเครื่องแม่ข่าย (Server) ที่ให้บริการการพัฒนาระบบสารสนเทศภายในองค์กร ทำงานบนระบบปฏิบัติการ Linux โดยติดตั้งซอฟต์แวร์ CentOS เวอร์ชัน 6.5 RedHat Enterprise Linux (RHEL) และติดตั้งชุดโปรแกรมเพิ่ม (Package) ประกอบด้วย PHP เวอร์ชัน 5.4 , MySQL เวอร์ชัน 5.5 , PhpMyAdmin เวอร์ชัน 4.2 , Apache เวอร์ชัน 2.2.15

ภาพที่ 6-27 ขอบเขตของซอฟต์แวร์

1.5 ขอบเขตของงานสหกิจศึกษาและข้อจำกัดของปัญหา

ได้รับมอบหมายให้ดำเนินการพัฒนาระบบสารสนเทศ ระบบติดตามและตรวจสอบนิสิตระดับบัณฑิตศึกษา ในส่วนของการจัดการข้อมูลนิสิต ข้อมูลบุคลากรภายนอก ข้อมูลพื้นฐานระบบ ซึ่งจะเป็นส่วนที่นำไปใช้ต่อไปในการพัฒนาส่วนการทำงานต่างๆของระบบ

1) การจัดการข้อมูลนิสิตรายบุคคล

เจ้าหน้าที่สามารถเพิ่ม และแก้ไขข้อมูลนิสิต โดยมีรายละเอียดดังนี้

- สามารถเลือก คำนำหน้า (ภาษาไทย) ในภาพแบบของดรอปดาวน์

(Drop down list) เพื่อให้ผู้ใช้เลือกข้อมูล 1 รายการ

- สามารถป้อนข้อมูล ชื่อ (ภาษาไทย)
- สามารถป้อนข้อมูล นามสกุล (ภาษาไทย)
- สามารถเลือก คำนำหน้า (ภาษาอังกฤษ) ในภาพแบบของดรอปดาวน์

(Drop down list) เพื่อให้ผู้ใช้เลือกข้อมูล 1 รายการ

ภาพที่ 6-28 แผนการดำเนินงาน

- งานที่ได้รับมอบหมาย
- ช่วงเวลาที่ดำเนินงาน

ตารางที่ 1-1 แผนการปฏิบัติงาน ช่วงที่ 1					
หัวข้องาน		เดือนที่ 1		เดือนที่ 2	
การพัฒนาระบบเพื่อสนับสนุนกิจกรรมของสถาบันบัณฑิตศึกษา จุฬารักษ์ งานประชุมวิชาการนานาชาติในสาขาอินทรีย์เคมีของ นักศึกษาระดับบัณฑิตศึกษาในภูมิภาคเอเชีย ครั้งที่ 4					
1	เข้าร่วมประชุมกับผู้ที่เกี่ยวข้อง 1.1) ประชุมกับทีมพัฒนาซอฟต์แวร์ 1.2) ประชุมกับฝ่ายจัดทำ Abstract book	↔			
2	ออกแบบระบบส่วนของ interface	↔			
3	พัฒนาระบบในส่วน of ข้อมูลรายละเอียด 3.1) ส่วนของ Abstract Submission 3.2) ส่วนของข้อมูลตารางกิจกรรมการประชุม 3.3) ส่วนของข้อมูล Venue/Excursion 3.4) ส่วนของข้อมูล Accomodation 3.5) ส่วนของข้อมูล Contact	↔	↔		
4	ตรวจสอบความถูกต้องของเว็บไซต์		↔		
5	แนะนำ และแก้ไขปัญหาการใช้งานเว็บไซต์			↔	↔

ดัชนีชี้วัดความสำเร็จ

- ความสามารถในการด้านวิเคราะห์
- ความสามารถในการเขียนโปรแกรม

ภาพที่ 6-29 ความสามารถในการด้านวิเคราะห์

1.7.1 ความสามารถในการด้านวิเคราะห์

- 1) ประเมินจากเอกสารความต้องการของระบบ ขั้นตอนในการวิเคราะห์ระบบ
- 2) ประเมินจากการสรุปผลที่เป็นลำดับขั้นตอน แสดงขอบเขตการทำงานได้อย่างชัดเจน

ภาพที่ 6-30 ความสามารถในการเขียนโปรแกรม

1.7.2 ความสามารถในการเขียนโปรแกรม

- 1) ประเมินจากความสามารถในการเขียนโปรแกรมสอดคล้องกับแผนภาพที่ได้ออกแบบวิธีแก้ปัญหาหรือขั้นตอนกระบวนการทำงานไว้
- 2) ประเมินจากการใช้งานร่วมกันของฟังก์ชัน และตัวแปรร่วมกันภายในระบบๆ เพื่อเป็นการลดขั้นตอนการเขียนโปรแกรมซ้ำซ้อน และลดความผิดพลาดของข้อมูล เป็นต้น
- 3) ประเมินจากการทำงานของระบบๆ ต้องทำงานได้ตรงตามข้อกำหนดเอกสารความต้องการ หรือภายใต้ขอบเขตการทำงานที่ออกแบบไว้

รายละเอียดของการปฏิบัติงาน

- นิยามคำศัพท์ของคำสำคัญ
- ทฤษฎีที่เกี่ยวข้อง
- งานวิจัย หรือบทความที่เกี่ยวข้อง

ภาพที่ 6-31 นิยามคำศัพท์ของคำสำคัญ

2.1 นิยามคำศัพท์ของคำสำคัญ

ในส่วนนี้จะให้คำจำกัดความของคำศัพท์ที่เกี่ยวข้องกับเอกสารฉบับนี้ ดังปรากฏในตารางที่ 2-1
ตารางที่ 2-1 ตารางแสดงคำศัพท์ระบบ Content Management System (CMS)

ลำดับ	คำศัพท์	ความหมาย
1	front-end	ส่วนที่แสดงผลให้กับผู้เข้าชมเว็บไซต์ หรืออีกนัยหนึ่งก็คือเนื้อหาของเว็บไซต์นั่นเอง
2	back-end	ส่วนการจัดการเนื้อหา รวมถึงโครงสร้างของเว็บไซต์ หรือเรียกอีกชื่อหนึ่งว่าส่วน administrator
3	blog	เป็นรูปแบบเว็บไซต์ประเภทหนึ่ง ซึ่งถูกเขียนขึ้นในลำดับที่เรียงตามเวลาในการเขียน ซึ่งจะแสดงข้อมูลที่เขียนล่าสุดไว้แรกสุด บล็อกโดยปกติจะประกอบด้วยข้อความ ภาพ ลิงก์

ภาพที่ 6-32 ทฤษฎีที่เกี่ยวข้อง

2.2 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องกับการพัฒนาเว็บไซต์การประชุมวิชาการนานาชาติ ครั้งที่ 4 เป็นเรื่องของเทคนิค และเครื่องมือสำคัญที่ใช้ในการพัฒนาให้เว็บไซต์ตอบสนองกับผู้ใช้ให้มากที่สุด อีกทั้งรวมถึงการทำระบบจัดการฝึกอบรม ส่วนใหญ่แล้วผู้พัฒนาต้องมีความรู้ความเข้าใจเรื่องของการพัฒนาบุคลากรด้วยการจัดโครงการฝึกอบรมภายในองค์กรเสียก่อน นอกจากนี้กระบวนการฝึกอบรม และการบริหารงานฝึกอบรมก็จำเป็นที่จะต้องศึกษาค้นคว้าเพิ่มเติมด้วย ทั้งนี้ได้มีการยกบทความที่เกี่ยวข้องกับระบบจัดการฝึกอบรมมาด้วย

2.2.1 ทฤษฎีการศึกษาการทำงานของระบบ Content Management System (CMS)

CMS ย่อมาจาก Content Management System เป็นระบบที่นำมาช่วยในการสร้างและบริหารเว็บไซต์แบบสำเร็จรูป โดยในการใช้งาน CMS นั้นผู้ใช้งานไม่จำเป็นต้องมีความรู้ในด้านการเขียนโปรแกรม ก็สามารถสร้างเว็บไซต์ได้ด้วยตนเอง ตัวของ CMS จะมีโปรแกรมแถมมาให้ และสามารถเพิ่มเติมเองได้มากมายเช่น กระดาษค้นหา ระบบจัดการป้ายโฆษณา ระบบนับจำนวนผู้ชม แม้แต่กระทั่งตระกร้าสินค้า และอื่นๆอีกมากมาย

CMS เป็นเหมือนโปรแกรม โปรแกรมหนึ่ง ที่มีผู้พัฒนามาจากภาษาคอมพิวเตอร์ที่ใช้ในเว็บไซต์เช่น PHP , Python , ASP , JSP ซึ่งในปัจจุบันมีผู้พัฒนา CMS เพิ่มขึ้นมากมายอย่างเช่น Mambo Joomla และ Word press

ภาพที่ 6-33 งานวิจัย หรือบทความที่เกี่ยวข้อง

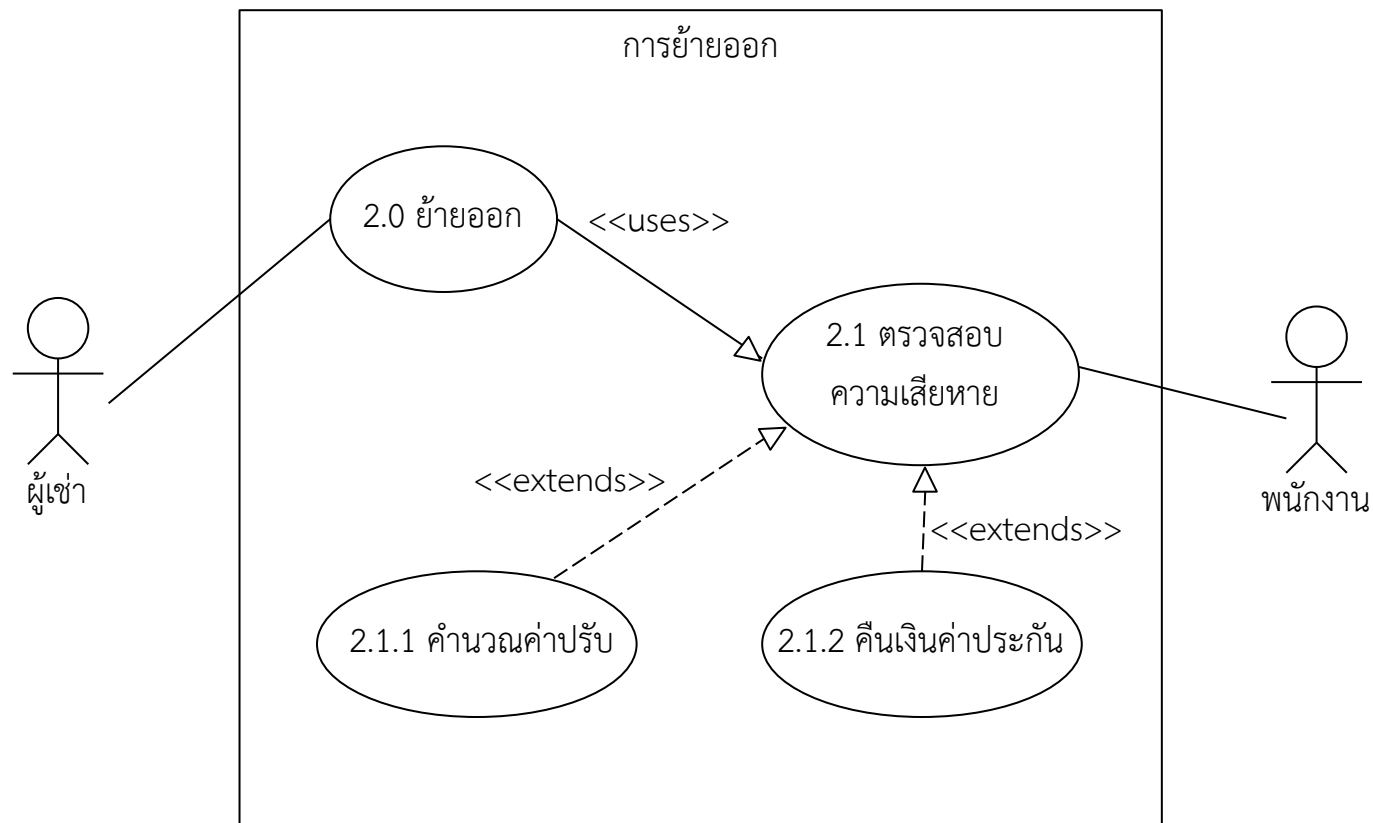
2.3 งานวิจัยหรือบทความที่เกี่ยวข้อง

ในการศึกษาเพิ่มเติมได้หาข้อมูลจากบทความ ข้อบังคับ กฎระเบียบต่างๆ ในการทำวิทยานิพนธ์ ดุษฎีนิพนธ์ และงานนิพนธ์ ซึ่งเป็นหน่วยงานของคณะวิทยาศาสตร์ มหาวิทยาลัยบูรพา เช่น แนวปฏิบัติการกำหนดเลขรหัสวิชาและชื่อวิชาของหลักสูตรระดับบัณฑิตศึกษา การทำวิทยานิพนธ์ของนิสิตปริญญาโท คณะวิทยาศาสตร์ มหาวิทยาลัยบูรพา เป็นต้น

แผนภาพต่าง ๆ

- Use case diagram
- Use case description
- Activity diagram
- Class diagram
- Entity Relationship diagram (ER diagram)

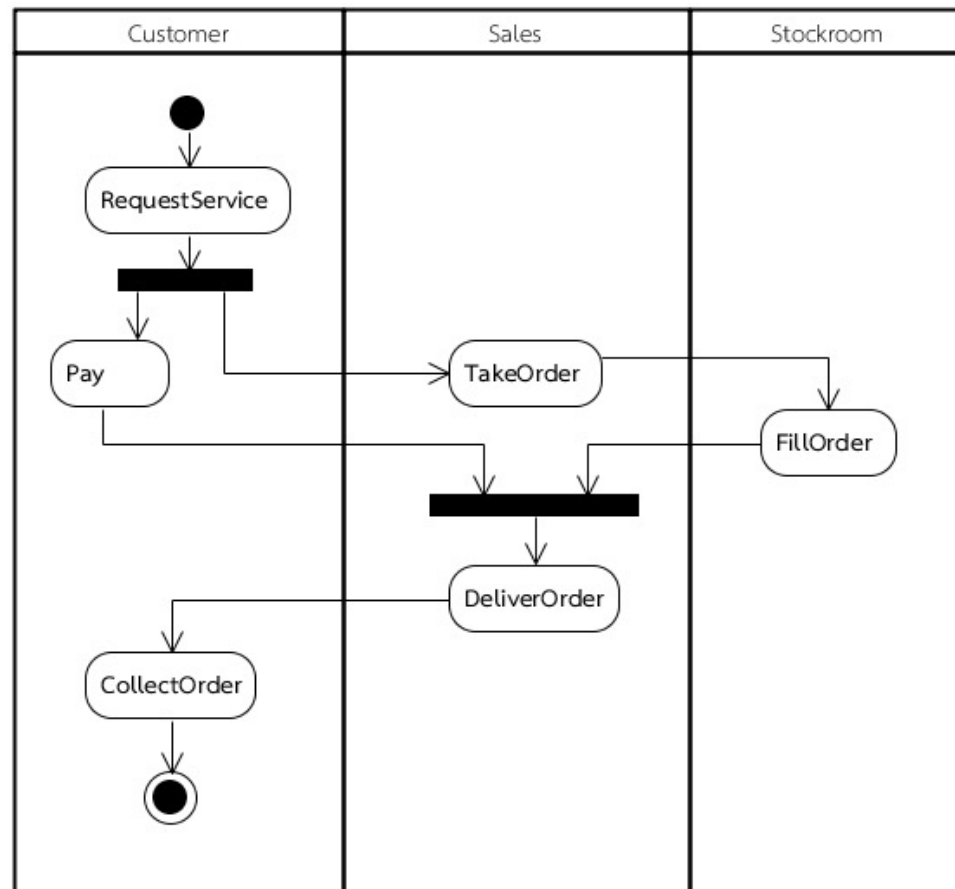
ภาพที่ 6-34 Use case diagram



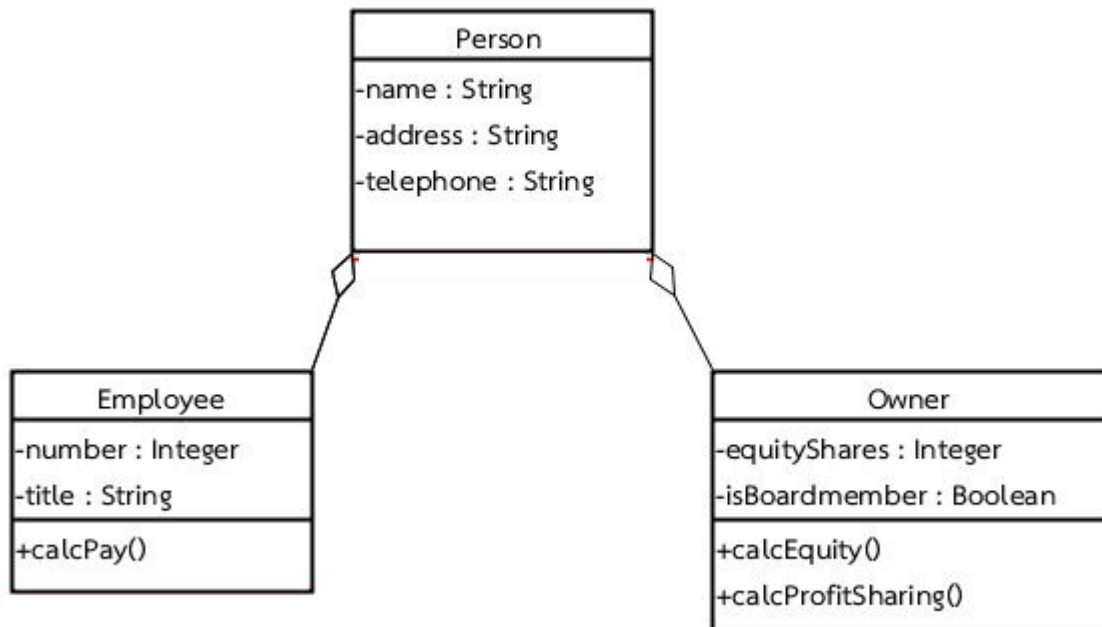
ภาพที่ 6-35 Use case description

Use Case ID	UC02	
Use Case Name:	ดูประวัติความเป็นมา	
Brief Description:	ใช้ในการดูประวัติความเป็นมาของหोजดหมายเหตุ	
Triggering Event:	ต้องมีข้อมูลของประวัติความเป็นมาอยู่ในระบบ	
Actors:	ผู้ใช้งาน ผู้ดูแลระบบ ผู้ดูแลระบบสูงสุด	
Preconditions:	-	
Postconditions:	แสดงข้อมูลประวัติความเป็นมาของหोजดหมายเหตุ	
Flow of Events:	Actor	System
	1. เข้าสู่หน้าแรกของระบบ 3. เลือกเมนู “ความเป็นมา”	2. ระบบแสดงข้อมูลหน้าแรก 4. ระบบแสดงข้อมูลของประวัติความเป็นมาของหोजดหมายเหตุ
Exceptions:	-	

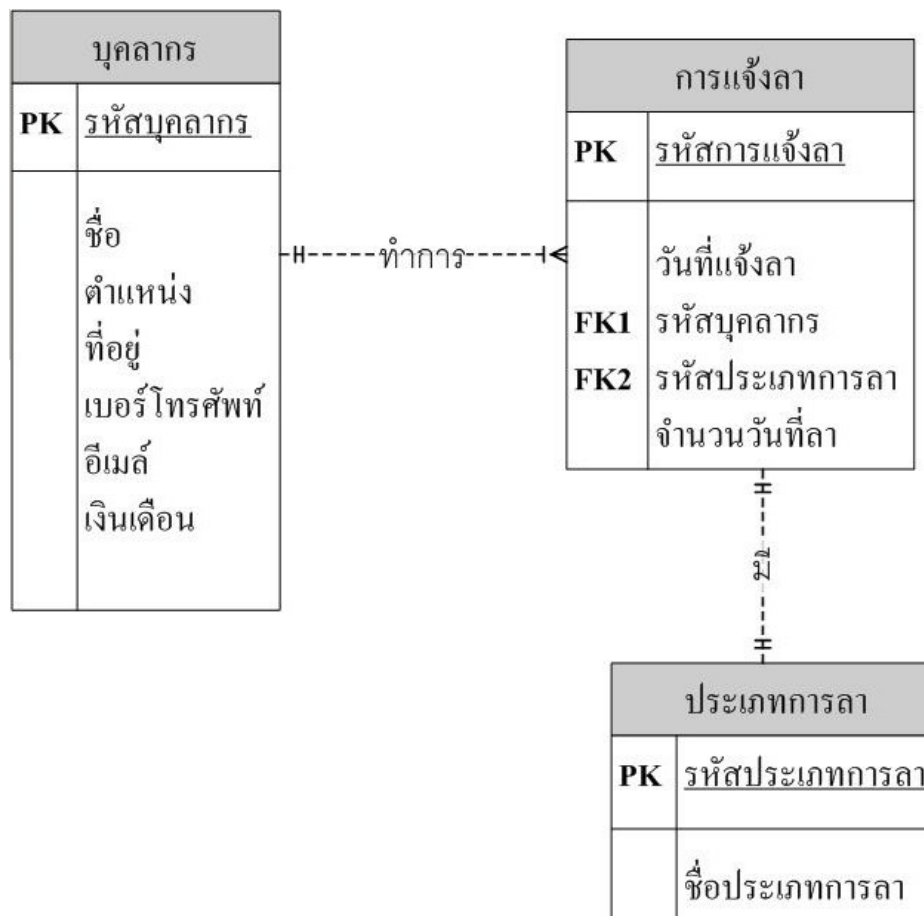
ภาพที่ 6-36 Activity diagram



ภาพที่ 6-37 Class diagram



ภาพที่ 6-38 Entity Relationship diagram



ภาพที่ 6-39 บรรณานุกรม

บรรณานุกรม

- [1] นฤชา มุสิกธิมณี. การเขียนรายงานสหกิจศึกษา รายงานปฏิบัติงานสหกิจศึกษา. 2557.
- [2] CLBS JOB. Agile Development. [ออนไลน์]. เข้าถึงได้จาก :
<http://www.clbs.co.th/th/jpbs/agile-software.html>. (วันที่สืบค้นข้อมูล: 12 ธันวาคม 2557).
- [3] Hellomyweb. PHP. [ออนไลน์]. เข้าถึงได้จาก :
<http://www.hellomyweb.com/index.php/main/content/135>. (วันที่สืบค้นข้อมูล: 15 ธันวาคม 2557).

ภาพที่ 6-40 ระดับชั้นที่ใช้การเขียนหัวข้อย่อย

- 1.1.1 xxxxxxxxxxxxxxxxxxxxxxxx
 - 1) xxxxx
 - 1.1) xxxxx
 - 1.1.1) xxxxx
 - xxxxx

1.3 ขอบเขตของโครงการ

ขอบเขตที่ต้องการพัฒนาระบบติดตามและตรวจสอบนิสิตระดับบัณฑิตศึกษา ได้แก่ การจัดการแบบฟอร์มวิทยานิพนธ์/คุษฎีนิพนธ์ การจัดการอนุมัติ การตรวจสอบสถานะการทำวิทยานิพนธ์/คุษฎีนิพนธ์ การจัดการข้อมูลทุน การจัดการข้อมูลตีพิมพ์ และรายงานการได้รับทุน ซึ่งแจกแจงรายละเอียด ดังต่อไปนี้

1.3.1 จัดการแบบฟอร์มวิทยานิพนธ์/คุษฎีนิพนธ์ และงานนิพนธ์

นิสิตและเจ้าหน้าที่ฝ่ายบัณฑิตศึกษาสามารถจัดการข้อมูลแบบฟอร์มวิทยานิพนธ์/คุษฎีนิพนธ์ และงานนิพนธ์ได้ ตามแบบฟอร์มที่กำหนด

- 1) วท 03 รายงานผลการสอบเค้าโครงวิทยานิพนธ์/คุษฎีนิพนธ์ เจ้าหน้าที่บันทึกข้อมูล

และอัปโหลดแบบฟอร์มเข้าระบบ

1.1) สามารถเลือกรายชื่อนิสิตที่ต้องการบันทึกรายงานผลการสอบเค้าโครงวิทยานิพนธ์/คุษฎีนิพนธ์ได้

1.2) สามารถแสดงข้อมูลนิสิตได้

1.3) สามารถแนบไฟล์ได้มากกว่า 1 ไฟล์ และมีขนาดไฟล์ไม่เกิน 4 เมกะไบต์ ซึ่งระบบสามารถรองรับรูปแบบไฟล์ ดังต่อไปนี้

1.3.1) รูปแบบไฟล์เอกสาร มีรูปแบบไฟล์ ได้แก่

- .docx
- .doc

1.3.2) แผ่นตารางทำการ (Spreadsheet) ได้แก่

- .xlsx
- .xls



ภาพที่ 6-41 ระยะห่างในการย่อหน้า และเว้นวรรค

14) วท 14 แบบคำร้องทั่วไปเกี่ยวกับการทำวิทยานิพนธ์/ดุขฎินิพนธ์ นิสิตดาวนโหลดแบบฟอร์มจากระบบ และส่งให้เจ้าหน้าที่บันทึกข้อมูล และอัปโหลดแบบฟอร์มเข้าระบบ

14.1) สามารถเลือก รายชื่อนิสิตที่ทำการส่งแบบคำร้องทั่วไปเกี่ยวกับวิทยานิพนธ์/ดุขฎินิพนธ์ ทีละหลายรายการ

14.2) สามารถแนบไฟล์ แบบคำร้องทั่วไปเกี่ยวกับวิทยานิพนธ์/ดุขฎินิพนธ์ ในรูปแบบของ .doc/.docx ขนาดไฟล์ทั้งหมดไม่เกิน 4 เมกกะไบต์ ได้ 1 ไฟล์

15) วท 15 แบบคำร้องการย้ายสาขาวิชาของนิสิตระบบบัณฑิตศึกษา นิสิตดาวนโหลดแบบฟอร์มจากระบบ และส่งให้เจ้าหน้าที่บันทึกข้อมูล และอัปโหลดแบบฟอร์มเข้าระบบ

15.1) สามารถกรอก หลักสูตรใหม่ของนิสิต

15.2) สามารถเลือก รายชื่อนิสิตที่ต้องการบันทึกได้

15.3) สามารถแนบบแบบคำร้องการย้ายสาขาวิชาของนิสิตระบบบัณฑิตศึกษา ในรูปแบบของ .doc/.docx ขนาดไฟล์ทั้งหมดไม่เกิน 4 เมกกะไบต์ ได้ 1 ไฟล์

1 นิ้ว



0.5 นิ้ว



0.25 นิ้ว



การเขียนภาษาอังกฤษทุกคำขึ้นต้นด้วยตัวอักษรตัวใหญ่

- แผนภาพแสดงการทำงานของระบบที่มีรูปแบบ OOP (Object Oriented Programming)
- โดยใช้สัญลักษณ์ UML (Unified Modeling Language)
- ภาษา SQL (Structured Query Language)
- การพัฒนาซอฟต์แวร์ (Software Development Life Cycle: SDLC)

การใช้เครื่องหมาย จุลภาค (Comma) ในการเขียนประโยคภาษาอังกฤษ

- แยกของที่อยู่ในกลุ่มออกจากกัน
 - เช่น I can speak American English, French, German and Mandarin Chinese
- แยกส่วนขยาย หรือข้อความที่สามารถละได้ออกจากประโยค
 - เช่น *Mary walked to the party, but she was unable to walk home.*

การใช้เครื่องหมายไม่ยมก (๑)

- วรรคหน้า วรรคหลัง
 - ▣ ระบบสารสนเทศเพื่อสนับสนุนการผลิตบัณฑิต ประกอบไปด้วยระบบต่าง ๆ ได้แก่ ระบบการรับสมัคร (Admission) ระบบประเมินการศึกษา (Open AMS) ระบบทะเบียนและประมวลผลการศึกษา (E-regist) และระบบกิจการนักศึกษา (Open SA)

การใช้เครื่องหมายไปยาลน้อย (๑) และไปยาลใหญ่ (๑๒๑)

- ไปยาลน้อย (๑) วรรคหลัง
 - ▣ เช่น การปฏิบัติงานสหกิจศึกษา ณ ห้องปฏิบัติการวิจัยวิศวกรรมระบบสารสนเทศ คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา ห้องปฏิบัติการวิจัย^๑ ได้มอบหมายให้เจ้าหน้าที่ที่ปรึกษา คือ นายมานิชญ์ ใจกว้าง เป็นผู้ดูแลให้คำปรึกษา
- ไปยาลใหญ่ (๑๒๑) วรรคหน้า วรรคหลัง
 - ▣ เช่น ปัจจุบันนี้มีภาษาที่ใช้เขียนโปรแกรมจำนวนมาก เช่น ภาษาซี ภาษารูบี้ ภาษาจาวา ^{๑๒๑}

การใช้เครื่องหมายวงเล็บ ()












- วรรคหน้า วรรคหลัง
 - เช่น แผ่นตารางทำการ (Spreadsheet) หมายถึง แผ่นตารางที่ประกอบด้วยแนวตั้ง และแนวนอน ตัดกันเป็นช่องสี่เหลี่ยม แนวตั้งเรียกว่า สดมภ์ (Column) แนวนอนเรียกว่า แถว (Row)

คำสรรพนามในประโยค

- ไม่ควรใช้คำสรรพนาม ข้าพเจ้า ฉัน ผม เรา เขา เธอ มัน ท่าน คุณ เป็นต้น
 - เช่น การปฏิบัติงานสหกิจศึกษาในระยะเวลา 16 สัปดาห์ที่ผ่านมา ผมได้รับมอบหมายให้พัฒนาระบบจัดการโครงการซอฟต์แวร์
- ให้ใช้คำว่า ผู้ปฏิบัติงาน ผู้ดำเนินงาน นิสิตผู้ปฏิบัติงาน เป็นต้น

การใช้คำว่า “และ” สำหรับการเขียนประโยคภาษาไทย

- ใช้เมื่อต้องการอธิบายว่ามีรายการอะไรบ้าง เช่น
 - ฉันสามารถพูดภาษาอังกฤษ ไทย จีน **และ** ลาวได้
 - รูปแบบการประเพณีมี 2 ลักษณะ คือ การประเพณีทั่วไป**และ**การประเพณีบุคลากร
 - ในการสำเร็จการศึกษา สาขาวิชาวิศวกรรมซอฟต์แวร์ จะต้องผ่านการสหกิจศึกษา **และ**จัดทำโครงการงาน

- | โครงสร้างประเภทงบประมาณ | | | |
|--|----------------------|-------------------------------------|---|
| รายละเอียดประเภทงบประมาณ | | ใช้งาน | ตัวเลือก |
| <input checked="" type="checkbox"/> งบบุคลากร | + เพิ่มหมวดงบประมาณ | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> เงินเดือน | + เพิ่มประเภทรายจ่าย | <input checked="" type="checkbox"/> |  |
| เงินเดือนข้าราชการ | | <input checked="" type="checkbox"/> |  |
| เงินประจำตำแหน่ง | | <input checked="" type="checkbox"/> |  |
| เงินช่วยเหลือการครองชีพ | | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> ค่าจ้างประจำ | + เพิ่มประเภทรายจ่าย | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> ค่าจ้างชั่วคราว | + เพิ่มประเภทรายจ่าย | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> งบดำเนินงาน | + เพิ่มหมวดงบประมาณ | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> งบรายจ่ายลงทุน | + เพิ่มหมวดงบประมาณ | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> งบรายจ่ายอุดหนุน | + เพิ่มหมวดงบประมาณ | <input checked="" type="checkbox"/> |  |
| <input checked="" type="checkbox"/> งบรายจ่ายอื่น | + เพิ่มหมวดงบประมาณ | <input checked="" type="checkbox"/> |  |

ภาพที่ 4-32 หน้าจอเมนูจำแนกประเภทงบประมาณ

ภาพที่ 6-43 การใช้ตาราง

- ชื่อตารางจะอยู่ตำแหน่ง **ชิดซ้าย บนหัวตาราง**

ตารางที่ 2-1 ตารางแสดงคำศัพท์ที่ใช้ในระบบจัดการแผนยุทธศาสตร์และแผนปฏิบัติการประจำปี

ลำดับ	คำศัพท์	ความหมาย
1.	แผนยุทธศาสตร์	ทิศทางหรือแนวทางปฏิบัติตามพันธกิจและภารกิจ ให้สัมฤทธิ์ผลตามวิสัยทัศน์ และเป้าประสงค์ขององค์กร
2.	ปีงบประมาณ	ระยะตั้งแต่วันที่ 1 ตุลาคมของปีหนึ่ง ถึง 30 กันยายนของปีถัดไป และให้ใช้ปี พ.ศ. ที่ถัดไปนั้นเป็นชื่อสำหรับปีงบประมาณ

ภาพที่ 6-44 การอ้างอิงรูปภาพ หรือตาราง


Prabromarajchanok
 Institute Executive
 Information System

ม.บูร -
 ผู้รับคิชอบหลักสูตร
 วิทยาลัยพยาบาลบรมราชชนนีนครลำปาง

🏠 หน้าหลัก > ผู้รับคิชอบหลักสูตร > จัดการ Common data set

หลักสูตร : กรุณาเลือกหลักสูตร ปีประเมิน : กรุณาเลือกปีประเมิน แสดงข้อมูล

การจัดการ Common data set เกณฑ์ สกอ. ระดับหลักสูตร

กรุณาเลือกหลักสูตรและปีประเมิน

ภาพที่ 4-4 หน้าหลักการจัดการ Common data set

หน้าหลักการจัดการ Common data set ที่ปรากฏในภาพที่ 4-4 ประกอบไปด้วยเมนูเลือกหลักสูตร และปีประเมินเพื่อใช้เป็นข้อมูลตั้งต้นในการค้นหาข้อมูลที่เกี่ยวข้องกับการประเมิน ส่วนการอัปเดตข้อมูล แสดงขึ้นเมื่อเลือกหลักสูตรและปีประเมินแล้วเหนือข้อมูล Common data set

ภาพที่ 6-45 ตัวอย่างตาราง และคำอธิบายตาราง

ตารางที่ 3-2 Use Case Description การค้นหาปฏิทินหลักสูตรฝึกอบรม

Name	ค้นหาปฏิทินหลักสูตรฝึกอบรม
Use Case ID	UC02
Actor	Admin, บุคคลภายใน, บุคคลภายนอก
Description	เป็นยูสเคสสำหรับการเลือกค้นหาหลักสูตรฝึกอบรมในรูปแบบปฏิทิน
Pre-Condition	- มีข้อมูลหลักสูตรฝึกอบรม
Post-Condition	-

จากตารางที่ 3-2 Use Case Description ที่แสดงขั้นตอนการดำเนินการค้นหาปฏิทินหลักสูตรฝึกอบรม โดยมีผู้ใช้ประเภท Admin บุคคลภายใน และบุคคลภายนอก เป็นผู้เกี่ยวข้องในการค้นหาปฏิทินหลักสูตรฝึกอบรม

เอกสารอ้างอิง

1. Axel van Lamsweerde. Requirements Engineering From System Goals to UML Models to Software Specifications (Chapter 4), Wiley, ISBN 978-0-470-01270-3, 2009.
2. Ian F. Alexander and Richard Stevens. Writing Better Requirements (Chapter 5 – Chapter 7), Addison Wesley, ISBN 0-321-13163-0, 2002.
3. Image icon human-3D , www.mehrad-co.com
4. สไลด์ Lecture4 สืบค้นวันที่ 20 ธันวาคม 2556 เข้าถึงได้จาก <http://www.ics.uci.edu/~taylor/classes/113/Chapter%2004.pdf>