

บทที่ 5

การวิเคราะห์ความต้องการและแผนภาพยูสเคส

เกริ่นนำ

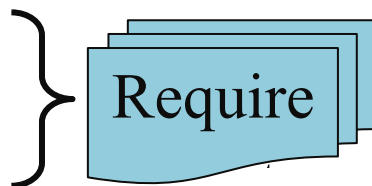
กระบวนการวิเคราะห์ระบบ (System Analysis Phase) คือขั้นตอนการพิจารณาที่มุ่งเน้นว่าจะอะไรหรือสิ่งใดที่ระบบจะต้องมีและต้องทำให้กับผู้ใช้โดยยังไม่เน้นว่าจะทำอย่างไร โดยในขั้นตอนนี้เป็นการหาความต้องการของผู้ใช้ ซึ่งเน้นที่กระบวนการวิเคราะห์ความต้องการของผู้ใช้ระบบ (Requirement Analysis Phase) เพื่อใช้ในการสร้างแบบจำลองหน้าที่การทำงานของระบบซอฟต์แวร์ จากมุมมองของผู้ใช้หรือระบบภายนอกซึ่งจะได้ผลลัพธ์เป็นแบบจำลองของความต้องการของผู้ใช้ระบบ (Requirement Model)

ความหมายของการวิเคราะห์

การวิเคราะห์ หมายถึง เป็นการแยกแยะสิ่งที่จะพิจารณาออกเป็นส่วนย่อยที่มีความสัมพันธ์กัน เพื่อทำความเข้าใจแต่ละส่วนให้แจ่มแจ้ง รวมทั้งการสืบค้นความสัมพันธ์ของส่วนต่าง ๆ เพื่อดูว่าส่วนประกอบปลีกย่อยนั้นสามารถเข้ากันได้หรือไม่ สัมพันธ์เกี่ยวเนื่องกันอย่างไร ซึ่งจะช่วยให้เกิดความเข้าใจต่อสิ่งหนึ่งสิ่งใดอย่างแท้จริง โดยพื้นฐานแล้ว การวิเคราะห์ถือเป็นทักษะที่มนุษย์ฝึกได้ (วิกิพีเดีย)

การวิเคราะห์ระบบ (Systems Analysis) เป็นการศึกษาลึกลงไปที่เกิดขึ้นในระบบงานปัจจุบัน (Current System) เพื่อออกแบบระบบการทำงานใหม่ (New System) นอกจากออกแบบสร้างระบบงานใหม่แล้ว เป้าหมายในการวิเคราะห์ระบบต้องการปรับปรุงและแก้ไขระบบงานเดิมให้มีทิศทางที่ดีขึ้น โดยก่อนที่ระบบงานใหม่ยังไม่นำมาใช้งาน ระบบงานที่ดำเนินการอยู่ในปัจจุบันเรียกว่า ระบบปัจจุบัน แต่ถ้าต่อมามีการพัฒนาาระบบใหม่และนำมาใช้งาน เราจะเรียกระบบปัจจุบันที่เคยใช้นั้นว่า ระบบเก่า (Old system) (วิกิพีเดีย , <https://th.wikipedia.org/wiki/การวิเคราะห์ระบบ>)

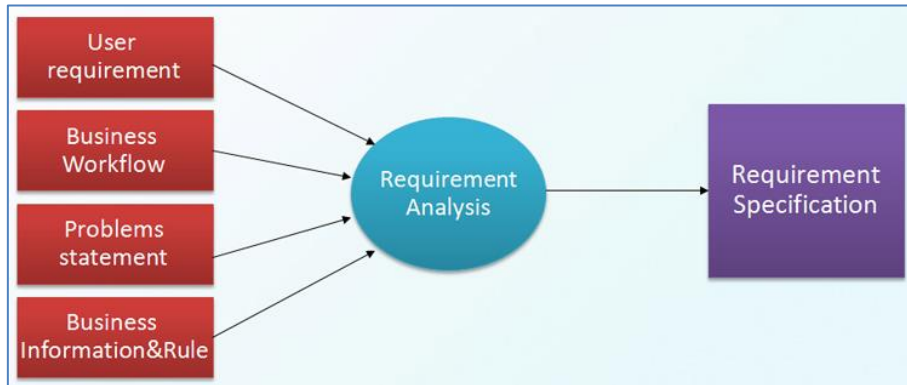
- 1) ระบบต้องขายสินค้าได้
- 2) บันทึกข้อมูลพนักงานได้
- 3)...



ความต้องการของผู้ใช้ระบบ

ความต้องการของผู้ใช้ระบบ (User requirements) หมายถึงคุณลักษณะในด้านต่าง ๆ ของระบบสารสนเทศที่กำลังจะทำการพัฒนาขึ้นเพื่อให้ระบบสามารถทำงานตอบสนองผู้ใช้ได้อย่างแท้จริงโดยแหล่งของความต้องการนั้นมาจากผู้ใช้ซึ่งนักวิเคราะห์ระบบจะต้องเป็นผู้สังเคราะห์ความต้องการนั้นจากข้อมูล

ต่าง ๆ ที่ได้รับมาจากผู้ใช้ แล้วจัดทำเป็นข้อกำหนดของความต้องการ (Requirement Specifications) เพื่อใช้เป็นเป้าหมายและขอบเขตของการพัฒนาระบบต่อไป



ภาพที่ 5.1 กระบวนการวิเคราะห์ความต้องการ

จากภาพที่ 5.1 อธิบายได้ว่าการวิเคราะห์ความต้องการคือกระบวนการในการวิเคราะห์เพื่อหาข้อกำหนดความต้องการของผู้ใช้ โดยจะต้องอาศัยข้อมูลในด้านต่าง ๆ เช่น ความต้องการจากผู้ใช้ กระแสงานทางธุรกิจ กรอบของปัญหา ข้อกำหนดทางธุรกิจหรือสารสนเทศทางธุรกิจ เป็นต้น ที่ได้รับมาจากผู้ใช้และองค์กรของผู้ใช้เพื่อทำการวิเคราะห์ จึงทำให้ได้มาซึ่งรายงานข้อกำหนดความต้องการ

ประเภทของผู้ใช้ระบบสารสนเทศ

ผู้ใช้งานระบบสารสนเทศ คือกลุ่มบุคคลที่มีส่วนได้ส่วนเสียหรือเป็นผู้ที่ใช้งานระบบซึ่งจะเป็นแหล่งที่มาของความต้องการนั้น ๆ โดยสามารถแบ่งผู้ใช้ออกเป็นกลุ่มต่าง ๆ ได้ดังต่อไปนี้

1. เจ้าของระบบ (System owners หรือ Sponsors) คือบุคคลที่มีส่วนได้ส่วนเสียจากการลงทุนสร้างหรือพัฒนาระบบสารสนเทศมากที่สุด อาจหมายถึงผู้จัดการ เจ้าของบริษัท หรือผู้บริหารองค์กรก็ได้

2. ผู้ใช้ภายใน (Internal users) ประกอบไปด้วยกลุ่มบุคคลต่าง ๆ ดังนี้

1) ผู้ใช้งานระดับปฏิบัติการ (End Users) คือผู้ใช้ที่ป้อนข้อมูลเข้าสู่ระบบโดยตรง ไม่จำเป็นต้องมีทักษะหรือความรู้มาก เน้นความถูกต้องและรวดเร็วของการป้อนข้อมูลเข้าสู่ระบบ เช่น เจ้าหน้าที่ฝ่ายทะเบียนนักศึกษา เจ้าหน้าที่จัดซื้อสินค้า เจ้าหน้าที่ฝ่ายบุคคล เป็นต้น

2) ผู้ใช้งานระดับเชี่ยวชาญ (Power Users) คือผู้ใช้ที่มีความรู้ความชำนาญเฉพาะด้าน สามารถใช้งานฟังก์ชันของระบบในส่วนที่มีความซับซ้อนได้ เช่น หัวหน้าเจ้าฝ่ายบุคคล ผู้จัดการฝ่ายลูกค้าสัมพันธ์ เป็นต้น

3) ผู้บริหาร (Administrators) คือผู้ที่ดูแลและควบคุมให้ระบบสามารถดำเนินการได้อย่างราบรื่นตามวัตถุประสงค์ที่ตั้งไว้

4) ผู้บริหารระดับสูง (Executive Users) คือผู้ใช้ที่ต้องการสารสนเทศมาเพื่อการตัดสินใจ และบริหารองค์กร

3. ผู้ใช้ภายนอก (External Users) คือผู้ใช้ซึ่งเป็นบุคคลหรือหน่วยงานภายนอกองค์กร แต่สามารถเข้าถึงบริการของระบบในองค์กรได้ เช่น บริษัทคู่ค้า กรมสรรพากร ธนาคาร เป็นต้น

เทคนิคการเก็บรวบรวมความต้องการ

การรวบรวมความต้องการ (Information Gathering) เป็นกระบวนการที่นักวิเคราะห์ระบบจะต้องให้ความสำคัญเป็นอย่างมากหากความต้องการที่รวบรวมมาไม่ครบหรือวิเคราะห์ความต้องการผิดพลาดไปย่อมก่อให้เกิดความเสียหายของระบบในอนาคตต่อไปได้ซึ่งระบบงานที่พัฒนาก็ย่อมตอบสนองความต้องการของผู้ใช้ระบบไม่ครบถ้วนหรือตอบสนองไม่ตรงตามความต้องการดังนั้นนักวิเคราะห์ระบบจะต้องคำนึงถึงความต้องการของผู้ใช้ระบบด้วยการยึดถือหลักความจริงที่ว่าระบบงานที่พัฒนาขึ้นมาเป็นของผู้ใช้ระบบเนื่องจากผู้ใช้ระบบจะเป็นผู้ที่ปฏิบัติงานกับระบบเพื่อใช้งานเป็นประจำ (รัฐพงศ์ ส่งเนียม, 2563)

นักวิเคราะห์ระบบจะต้องพบปะกับผู้คนหลายระดับด้วยกันตั้งแต่ระดับผู้บริหารจนถึงระดับผู้ปฏิบัติการเพื่อรวบรวมข้อมูลเกี่ยวกับความต้องการว่าต้องการอะไรบ้างในระบบใหม่ ทั้งนี้การรวบรวมความต้องการควรมีหลักการโดยกลยุทธ์ที่นักวิเคราะห์ระบบสามารถค้นหาความต้องการให้มีความชัดเจนขึ้นคือจะต้องมีแนวทางในการจัดการกับผู้คนที่เกี่ยวข้อง ซึ่งประกอบด้วย ใครอะไรเมื่อไร ที่ไหน ทำไม และอย่างไร โดยที่

ใคร (Who) ในส่วนงานดังกล่าว มีใครบ้างที่มีส่วนเกี่ยวข้องกับใครเป็นผู้ดำเนินงานในแต่ละขั้นตอนของระบบงานนี้และเขาเหล่านั้นทำไมจึงเป็นผู้เหมาะสมกับการปฏิบัติงานในกระบวนการนี้มีใครคนอื่นบ้างหรือเปล่าที่ทำงานได้ประสิทธิภาพดีกว่า

อะไร (What) อะไรคือสิ่งที่ทำให้เกิดปัญหาในระบบที่ต้องการคือระบบอะไร มีฟังก์ชันการทำงานอะไรบ้างทำไมขั้นตอนนี้จึงมีความจำเป็นต้องทำ

เมื่อไร (When) ระบบติดตั้งได้เมื่อไรเจ้าของธุรกิจพร้อมที่จะสนับสนุนเมื่อไรทดสอบระบบใหม่ได้เมื่อไรแล้วตอนนี้ดำเนินการได้เมื่อไรทำไมต้องดำเนินการเวลานี้มีเวลาอื่นที่เหมาะสมกว่าหรือไม่

ที่ไหน (Where) สถานที่แห่งไหนที่จะนำระบบใหม่ไปใช้งานมีการทำงานกันที่ไหนทำไมจึงเลือกสถานที่นี้ทำงาน หากทำที่อื่นคิดว่าจะดีกว่าสะดวกกว่าหรือไม่

ทำไม (Why) ทำไมต้องพัฒนาระบบใหม่และทำไมถึงเชื่อว่าระบบใหม่สามารถแก้ไขปัญหให้กับเขาได้

อย่างไร (How) ขั้นตอนการดำเนินงานจะต้องทำอย่างไรทำไมต้องทำแบบนี้มีวิธีอื่นที่น่าจะดีกว่านี้หรือไม่ที่จะทำให้งานออกมามีประสิทธิภาพดีกว่าค่าใช้จ่ายถูกกว่า

อย่างไรก็ตามการรวบรวมความต้องการผู้ใช้ระบบถือเป็นส่วนหนึ่งที่สำคัญทีเดียวในการให้ข้อมูลแก่นักวิเคราะห์ระบบทั้งนี้ นักวิเคราะห์ระบบคงไม่ใช่ทำงานอยู่ในห้องและขีดเขียนความต้องการขึ้นมาตาม

อำเภอใจแต่นักวิเคราะห์ระบบจะต้องออกนอกสถานที่เพื่อพบปะและสนทนากับผู้ใช้ระบบระดับต่างๆเพื่อให้รับทราบถึงปัญหา และนำมาประกอบการวิเคราะห์ระบบเช่นขั้นตอนการดำเนินงานที่ปัจจุบันดำเนินการอยู่ เป็นอย่างไรปัญหาที่เกิดขึ้นในแต่ละขั้นตอนมีอะไรบ้างรวมถึงรายละเอียดงานและเอกสารอื่นๆ ที่เกี่ยวข้อง เป็นต้นซึ่งงานดังกล่าวเป็นงานที่ผู้ใช้ระบบได้ปฏิบัติเป็นประจำทุกวันจึงทำให้ผู้ใช้ระบบมีประสบการณ์และสามารถมองลึกถึงรายละเอียดของปัญหาได้เป็นอย่างดีรวมถึงผู้ใช้ระบบอาจเสนอแนวทางในการแก้ไขปัญหาให้แก่ นักวิเคราะห์ระบบก็ได้ดังนั้น การได้รับความร่วมมือที่ดีจากผู้ใช้ระบบย่อมก่อให้เกิดการวิเคราะห์ข้อมูลเพื่อแก้ไขปัญหาได้ตรงตามวัตถุประสงค์มากยิ่งขึ้นดังนั้นคุณสมบัติในบทบาทและหน้าที่ของผู้ใช้ที่ดี จึงประกอบด้วย (นัฐพงศ์ ส่งเนียม, 2563)

1. ผู้ใช้ที่ดีควรอธิบายขั้นตอนการทำงานของระบบปัจจุบันที่ดำเนินการอยู่ได้
2. ผู้ใช้ที่ดีสามารถชี้แจงปัญหาต่าง ๆ ที่เกิดขึ้นในระบบได้
3. ผู้ใช้ที่ดีสามารถระบุความต้องการในระบบใหม่ได้
4. ผู้ใช้ที่ดีควรจัดเตรียมเอกสาร หรือรายงานที่เกี่ยวข้องให้กับนักวิเคราะห์ระบบได้
5. ผู้ใช้ที่ดีควรให้ความร่วมมือแก่นักวิเคราะห์ระบบ
6. ผู้ใช้ที่ดีควรมีส่วนร่วมต่อโครงการพัฒนาระบบ

ในการเก็บรวบรวมข้อมูลหรือความต้องการของระบบนั้นสามารถทำได้หลายวิธีด้วยกันดังนี้

- 1) การสัมภาษณ์ (Interview) นิยมใช้มากที่สุดการสัมภาษณ์แบ่งออกเป็น 2 รูปแบบด้วยกัน คือ
 1. การสัมภาษณ์แบบไม่มีโครงสร้าง เป็นการกำหนดคำถามก่อนว่าจะถามเกี่ยวกับเรื่องอะไร ซึ่งอาจมีหัวข้อคำถามอยู่ในใจและมักจะสัมภาษณ์ในลักษณะการพูดคุยสนทนาอย่างไรก็ตามการสัมภาษณ์ด้วยวิธีนี้ไม่เหมาะสมต่อการวิเคราะห์และออกแบบระบบ
 2. การสัมภาษณ์แบบมีโครงสร้างเป็นรูปแบบการสัมภาษณ์ที่มีการกำหนดคำถามเพื่อการสัมภาษณ์โดยเฉพาะและมีความชัดเจนคำถามที่ใช้อาจเป็นคำถามปลายเปิดเพื่อเปิดโอกาสให้คู่สนทนาสามารถแสดงความคิดเห็นหรือเป็นคำถามปลายปิดที่มีตัวเลือกให้ตอบหรืออาจเป็นสองแบบผสมผสานกันก็ได้ (ที่มา: http://www.pttc.ac.th/home/BcomPttc/e-learning/การวิเคราะห์ออกแบบระบบ/SA1-5/SA4/SA_4.html)
- 2) การแสดงลำดับเหตุการณ์ (Scenario) คือการเตรียมคำถามตามลำดับงานของผู้ใช้
- 3) สร้างต้นแบบ (Prototype) เช่น ออกแบบจอภาพบนกระดาษเพื่อทดสอบการยอมรับความต้องการในเบื้องต้น
- 4) การประชุม (Meeting) เป็นการเรียกกลุ่มบุคคลที่เกี่ยวข้องมาประชุมเพื่อขอความคิดเห็นและความต้องการ
- 5) การสังเกต (Observation) โดยตรวจสอบสภาพแวดล้อมการทำงานของผู้ใช้เป็นวิธีการที่ดีแต่มีค่าใช้จ่ายสูง

กระบวนการวิเคราะห์ความต้องการ

กระบวนการวิเคราะห์ความต้องการนั้นมีขั้นตอนดังต่อไปนี้

- ขั้นตอนที่ 1. เก็บรวบรวมข้อมูลที่เป็นข้อเท็จจริงต่าง ๆ (Data Gathering)
- ขั้นตอนที่ 2. วิเคราะห์เพื่อระบุถึงความต้องการต่าง ๆ (Requirement Identification)
- ขั้นตอนที่ 3. คัดเลือกส่วนที่เป็นสาระสำคัญและอยู่ในขอบเขตการพัฒนา (Requirement Selection)
- ขั้นตอนที่ 4. จัดจำแนกและจัดโครงสร้างของความต้องการ (Requirement Classification and Structuring)
- ขั้นตอนที่ 5. จัดลำดับความสำคัญและตกลงเจรจา (Prioritization and Negotiation)
- ขั้นตอนที่ 6. ตรวจสอบความถูกต้อง (Requirement Validation)
- ขั้นตอนที่ 7. จัดทำเป็นข้อกำหนดของความต้องการ (Requirement Specification)

การระบุความต้องการ

การระบุความต้องการ (Requirement Identification) หมายถึง การหาความต้องการจากผู้ใช้งานระบบโดยในการระบุความต้องการนั้นจำเป็นต้องกำหนดแต่ละความต้องการให้เป็นอิสระในตัวเองซึ่งจะต้องไม่ผสมปะปนกับความต้องการอื่น ๆ และความต้องการนั้นต้องสามารถทำการทดสอบได้ในภายหลัง โดยความต้องการจะต้องสามารถจัดการดังต่อไปนี้ได้

- 1) จัดกลุ่มของความต้องการ (Grouped) เช่น จัดกลุ่มตามมุมมองของผู้ใช้ เป็นต้น
- 2) จัดลำดับความสำคัญ (Prioritized) ของความต้องการ
- 3) กำหนดในระดับของนามธรรมและรายละเอียด (Level of Abstraction)
- 4) กำหนดความสัมพันธ์ในเชิงการขึ้นอยู่กับกัน (Dependency)
- 5) ความต้องการแต่ละความต้องการควรมีรหัสที่เป็นเอกเทศใช้ในการอ้างอิงได้ (Identification)
- 6) ควรแสดงด้วยประโยคพรรณนา (Descriptive) ที่เรียบง่ายตรงไปตรงมา ไม่ควรอธิบายด้วยประโยคที่ซับซ้อนจนไม่สามารถถอดใจความได้

ประเภทของความต้องการ

ความต้องการนั้นสามารถแบ่งออกเป็น 2 ประเภทใหญ่ๆ ได้ดังนี้

1. **ความต้องการเชิงฟังก์ชัน (Functional Requirements)** คือการระบุถึงบริการหลักที่ระบบสามารถทำได้ (Statements of Services) เพื่อให้การประยุกต์ใช้สามารถเป็นไปตามวัตถุประสงค์ ตัวอย่างเช่น ระบบสามารถบันทึกการสั่งซื้อสินค้าของลูกค้าได้ระบบสามารถออกรายงานสรุปยอดขายประจำเดือนได้ระบบจะต้องแก้ไขข้อมูลลูกค้าได้ เป็นต้น

2. ความต้องการไม่เป็นเชิงฟังก์ชัน (Non-functional Requirements) คือเงื่อนไขหรือข้อกำหนด (Constraint) ที่ได้กำหนดไว้ต่อการพัฒนาและการนำไปใช้ของระบบ ตัวอย่างเช่น ระบบจะต้องใช้งานบนระบบปฏิบัติการลินุกซ์ ระบบจะต้องมีความเร็วในการตอบสนอง (Response Time) ไม่ควรเกิน 3 วินาที และสามารถทำงานร่วมกับระบบงานเดิมได้ เป็นต้น

การจำแนกประเภทของความต้องการไม่เป็นเชิงฟังก์ชัน

การจำแนกประเภทของความต้องการไม่เชิงฟังก์ชัน (Non-Functional Requirements) สำหรับข้อกำหนดความต้องการที่ไม่เป็นเชิงฟังก์ชันนั้นยังสามารถแบ่งออกเป็นประเภทย่อยได้ 3 ประเภทดังนี้

- ข้อกำหนดความต้องการที่ไม่เป็นเชิงฟังก์ชันที่เกี่ยวข้องกับตัวผลิตภัณฑ์ (Product Requirements) เช่น
 - ใช้งานได้อย่างสะดวก(Usability Requirements)
 - มีประสิทธิภาพดี(Efficiency Requirements) เช่น มีความถูกต้อง มีความเร็วในการประมวลผล เป็นต้น
 - มีความมั่นคงสูง (Reliability Requirements)
 - สามารถใช้งานในสภาพแวดล้อมที่ต่างกันได้ (Portability Requirements)
- ข้อกำหนดความต้องการที่ไม่เป็นเชิงฟังก์ชันที่เกี่ยวกับการบริหารจัดการ (Organizational Requirements) เช่น
 - สามารถส่งมอบได้ในเวลาที่กำหนด (Delivery Requirements)
 - ต้องสร้างด้วยวิธีการและเทคโนโลยีที่กำหนด (Implementation Requirements)
 - ต้องพัฒนาโดยยึดตามมาตรฐานของการพัฒนาที่กำหนด (Standard Requirements) เช่น ให้กระบวนการพัฒนามีมาตรฐานตาม ISO เป็นต้น
- ข้อกำหนดความต้องการที่ไม่เป็นเชิงฟังก์ชันที่เกี่ยวกับการเชื่อมต่อกับระบบภายนอก (External Requirements) เช่น
 - จะต้องรองรับการเชื่อมต่อจากภายนอกได้ (Interoperability Requirements)
 - จะต้องไม่ผิดศีลธรรม(Ethical Requirements)
 - จะต้องไม่ผิดกฎหมาย(Legislative Law Requirements)

ระดับของความต้องการ

ในการวิเคราะห์ความต่อนั้นการทำความเข้าใจระดับของความต่อนเป็นสิ่งสำคัญเพื่อให้สามารถวิเคราะห์ระบบได้ตรงตามความต่อนได้ โดยความต่อนนั้นสามารถแบ่งออกได้เป็น 2 ระดับใหญ่ ๆ ดังนี้

- ความต้องการของผู้ใช้ (User Requirements)

- 1) ถ้อยแถลงที่เป็นภาษาธรรมชาติ (Natural Language) ตลอดจนแผนภาพต่าง ๆ ที่แสดงให้ทราบถึงความสามารถในการทำงานของระบบและเงื่อนไขในการทำงานของระบบ
 - 2) ความต้องการของผู้ใช้จะต้องเขียนให้ผู้ใช้เข้าใจได้ง่ายเป็นหลัก
2. ความต้องการของระบบ (System Requirements)
- 1) คือการระบุถึงสิ่งที่ระบบควรมี เพื่อให้สามารถทำงานได้ตรงตามความต้องการของผู้ใช้
 - 2) ทำหลังจากได้ข้อกำหนดความต้องการของผู้ใช้มาแล้ว
 - 3) มักจะเขียนด้วยแบบจำลอง (Models) เพื่อแสดงองค์ประกอบของระบบในแง่มุมต่างๆ
 - 4) ใช้เป็นข้อกำหนดในการออกแบบต่อไป

ตัวอย่างการเขียนข้อกำหนดความต้องการ

ตัวอย่างที่ 5.1 ระบบการจองห้องพักโรงแรม

โรงแรมพระนครแกรนด์วิว เป็นโรงแรมที่ดำเนินกิจการให้เช่าห้องพัก โดยลูกค้าสามารถโทรมาจองห้องพักกับเจ้าหน้าที่ โดยระบุ ชื่อผู้จอง จำนวนห้องพัก วันที่เข้า จำนวนผู้เข้าพัก และจำนวนวันที่พัก โดยพนักงานจะทำตรวจสอบห้องว่าง ถ้ามีห้องพักรว่างก็จะทำการจองห้องพักให้กับลูกค้า เมื่อลูกค้ามาเข้าพักก็แจ้งชื่อผู้จองห้องพักให้กับพนักงาน สามารถวิเคราะห์ความต้องการของระบบได้ดังนี้

Req01: ระบบสามารถเก็บข้อมูลของลูกค้าได้

ข้อกำหนดความต้องการ (Requirement specification)

1. ข้อมูลลูกค้าจะต้องเก็บในฐานข้อมูล
2. ระบบจะไม่ยอมให้เก็บข้อมูลลูกค้า ถ้าขาดข้อมูลต่อไปนี้ เลขที่บัตรประจำตัว ชื่อ นามสกุล อีเมลล์
3. ระบบจะต้องตรวจสอบรูปแบบของ เลขที่บัตรประจำตัว ก่อนจัดเก็บ
4. ระบบจะต้องตรวจสอบรูปแบบของอีเมลล์ก่อนจัดเก็บ
5. ข้อมูลของลูกค้าจะต้องถูกจัดเก็บให้เป็นความลับและปลอดภัย
6. จะต้องสามารถเรียกดูได้ในภายหลังว่า พนักงานคนใดเป็นผู้บันทึกหรือแก้ไขข้อมูลลูกค้า ครั้งล่าสุด

Req02: ค้นหาข้อมูลห้องพัก{จากความต้องการของพนักงาน}

ข้อกำหนดความต้องการ (Requirement Specification)

- 1) ระบบจะต้องสามารถค้นหาว่าห้องพักใดว่างหรือไม่ว่าง
- 2) ระบบสามารถตรวจสอบได้ว่าห้องใด มีลูกค้าคนใดเข้าพัก
- 3) ระบบสามารถตรวจสอบได้ว่าห้องพักนั้นครบกำหนดออกวันไหน
- 4) ระบบสามารถตรวจสอบได้ว่าลูกค้าใช้บริการมินิบาร์ (Mini Bar) หรือไม่

5) ตรวจสอบว่ามีแม่บ้านทำความสะอาดแล้วหรือไม่

Req03: การจองห้องพัก{จากความต้องการของลูกค้า}

ข้อกำหนดความต้องการ (Requirement Specification)

- 1) ระบบจะต้องตรวจสอบข้อมูลการสมาชิกได้
- 2) ระบบจะต้องทำการบันทึกว่าข้อมูลผู้จอง ได้แก่ รหัส ชื่อ-นามสกุล เบอร์โทร ที่อยู่
- 3) ระบบจะต้องสามารถตรวจสอบได้ว่า มีห้องว่างหรือไม่ {uses Req08}
- 4) จะสามารถบันทึกข้อมูลการจองห้องพักได้ {uses Req09} โดยเก็บ วันที่ เวลา จำนวนห้อง จำนวนผู้เข้าพัก วันที่เข้าพัก วันที่ออกได้
- 5) ระบบสามารถตรวจสอบได้ภายหลังว่า พนักงานคนใดเป็นผู้รับการจอง
- 6) ถ้าเป็นสมาชิกจะมีส่วนลด 10 % {extend Req09}

Req04 : *สมัครสมาชิก* {จากความต้องการของลูกค้า}

ในการสมัครสมาชิก จะมีเงื่อนไขดังนี้

- 1) ระบบจะต้องตรวจสอบข้อมูลการสมาชิกได้
- 2) ระบบจะต้องทำการบันทึกว่าข้อมูลผู้สมัคร ได้แก่ รหัส ชื่อ-นามสกุล เบอร์โทร ที่อยู่
- 3) ระบบจะต้องสามารถตรวจสอบได้ว่า มีห้องว่างหรือไม่ {uses Req08}
- 4) จะสามารถบันทึกข้อมูลการจองห้องพักได้ โดยเก็บ วันที่ เวลา จำนวนห้อง จำนวนผู้เข้าพัก วันที่เข้าพัก วันที่ออก
- 5) ระบบสามารถตรวจสอบได้ภายหลังว่าพนักงานคนใดเป็นรับสมัคร
- 6) ถ้าเป็นสมาชิก จะมีส่วนลด 10 %

ตัวอย่างที่ 5.2 ในการพัฒนาระบบเช่าคินหนังสือออนไลน์ มีความต้องการอะไรบ้าง

- 1) Req01 : ค้นหาข้อมูลหนังสือ
- 2) Req02 : สมัครสมาชิกได้
- 3) Req03 : เช่าหนังสือได้
- 4) Req04 : ชำระเงิน
- 5) Req05 : คืนหนังสือได้
- 6) Req..*N* : จ่ายค่าปรับเกินกำหนด

Functional requirement

แนวทางการเขียนความต้องการด้วยภาษาธรรมชาติ

แนวทางหนึ่งที่ยิยมใช้ในการเขียนความต้องการคือ การเขียนโดยใช้ภาษาธรรมชาติ คือ เป็นสร้างรูปแบบมาตรฐานในการเขียน (Standard Format) โดยเป็นการใช้ภาษาที่ตรงไปตรงมาอย่างมีเหตุมีผล และไม่มี ความขัดแย้งกัน เน้นคำที่เป็นส่วนสำคัญของความต้องการนั้น ๆ และเลี่ยงการใช้ศัพท์ทางเทคนิคมากจนเกินไป (โดยเฉพาะศัพท์เทคนิคทางคอมพิวเตอร์) ข้อเสียของการเขียนความต้องการด้วยภาษาธรรมชาติก็คือ ในการเขียนความต้องการด้วยภาษาธรรมชาติมีข้อเสียหลายประการ เช่นการสื่อความหมายที่ไม่ดีหรือมีความกำกวมไม่ชัดเจน (Lack of Clarity) มีความสับสน (Confusion) มีความคลุมเครือ (Amalgamation) อย่างไรก็ตามยังมีวิธีการอธิบายความต้องการแบบอื่นที่นอกเหนือจากภาษาธรรมชาติ ในการเขียนอธิบายความต้องการอาจจะใช้วิธีแบบอื่นที่นอกเหนือจากภาษาธรรมชาติได้ เช่น ภาษาที่เป็นโครงสร้าง (Structured Language) แบบจำลองสัญลักษณ์ภาพ (Graphical Model) ตัวแบบหรือข้อกำหนดทางคณิตศาสตร์ (Formal Mathematical Specification) เป็นต้น

การตรวจสอบความถูกต้องของความต้องการ

การตรวจสอบความถูกต้องของความต้องการ (Requirement Validation) คือการตรวจสอบว่าความต้องการที่ได้มานั้นถูกต้องและตรงกับความต้องการของผู้ใช้อย่างแท้จริงหรือไม่โดยมีหลักการดังนี้

- 1) ความถูกต้อง (Validity) ความต้องการนั้นตรงกับที่ผู้ใช้ต้องการจริงหรือไม่ สามารถแก้ปัญหาให้ผู้ใช้ได้จริงหรือไม่
- 2) ความขัดแย้ง (Consistency) มีความขัดแย้งกันระหว่างความต้องการหรือไม่
- 3) ความครบถ้วน (Completeness) เป็นความต้องการที่ครบถ้วนของผู้ใช้ทุกคนหรือไม่
- 4) ความเป็นจริง (Realism) สามารถสร้างได้จริงตามความต้องการหรือไม่
- 5) ความตรวจสอบได้ (Verifiability) สามารถตรวจสอบได้หลังจากพัฒนาเสร็จแล้วหรือไม่

เทคนิคที่ใช้ในการหาความต้องการ

ในการหาความต้องการใช้งานระบบจากผู้ที่มีเทคนิคที่น่าสนใจอยู่ 2 อย่างดังนี้คือ

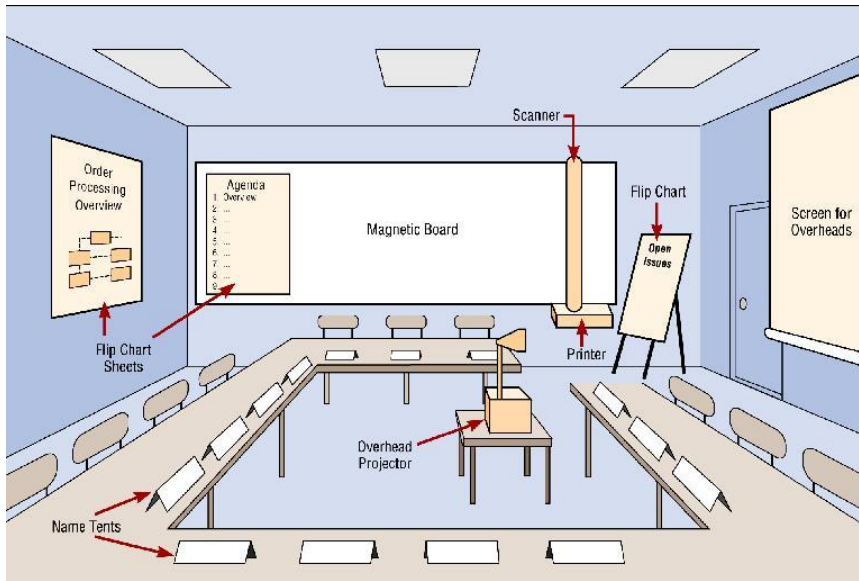
เทคนิคที่ 1. Joint Application Design (JAD)

เทคนิคที่ 2. Prototyping

1) เทคนิค Join Application Design (JAD)

เทคนิค JAD หรือ Join Application Design คือเทคนิคการกำหนดความต้องการของระบบโดยเน้นการรวบรวมบุคคลที่เกี่ยวข้องกับระบบหรือผู้มีส่วนได้ส่วนเสียกับระบบ (Stakeholders) มากำหนดความต้องการร่วมกันระหว่าง ผู้ใช้ เจ้าของระบบ ผู้บริหาร นักวิเคราะห์ระบบ และนักพัฒนาระบบ ซึ่งมีเป้าหมายคือหาความต้องการไปพร้อม ๆ กัน จากมุมมองของแต่ละคน (View Points) โดยการดำเนินการเป็นไปในลักษณะ

การประชุม จนสุดท้ายได้ข้อสรุปร่วมกันเป็นข้อกำหนดความต้องการซึ่งแสดงถึงคุณลักษณะของระบบที่
ต้องการ

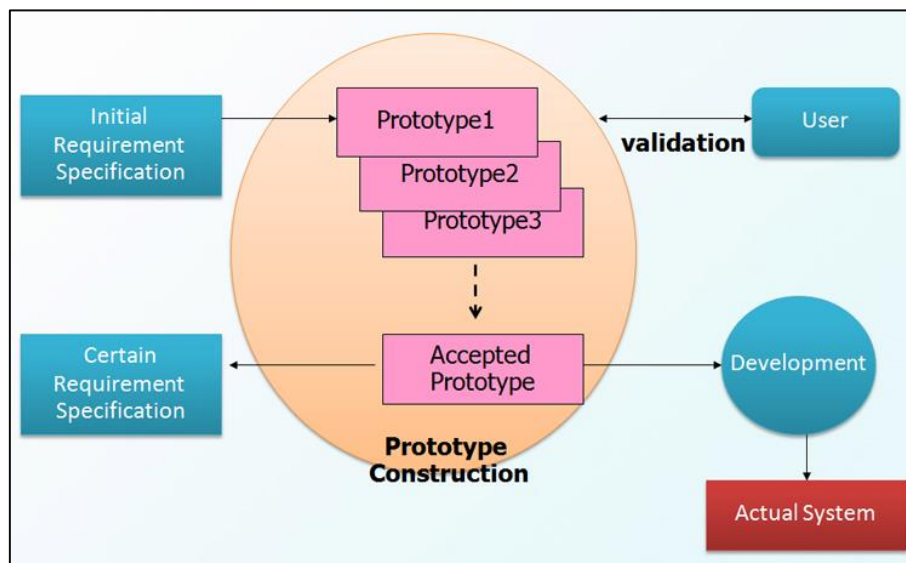


ภาพที่ 5.2 แสดงการรวบรวมข้อมูลด้วยวิธี JAD Technique

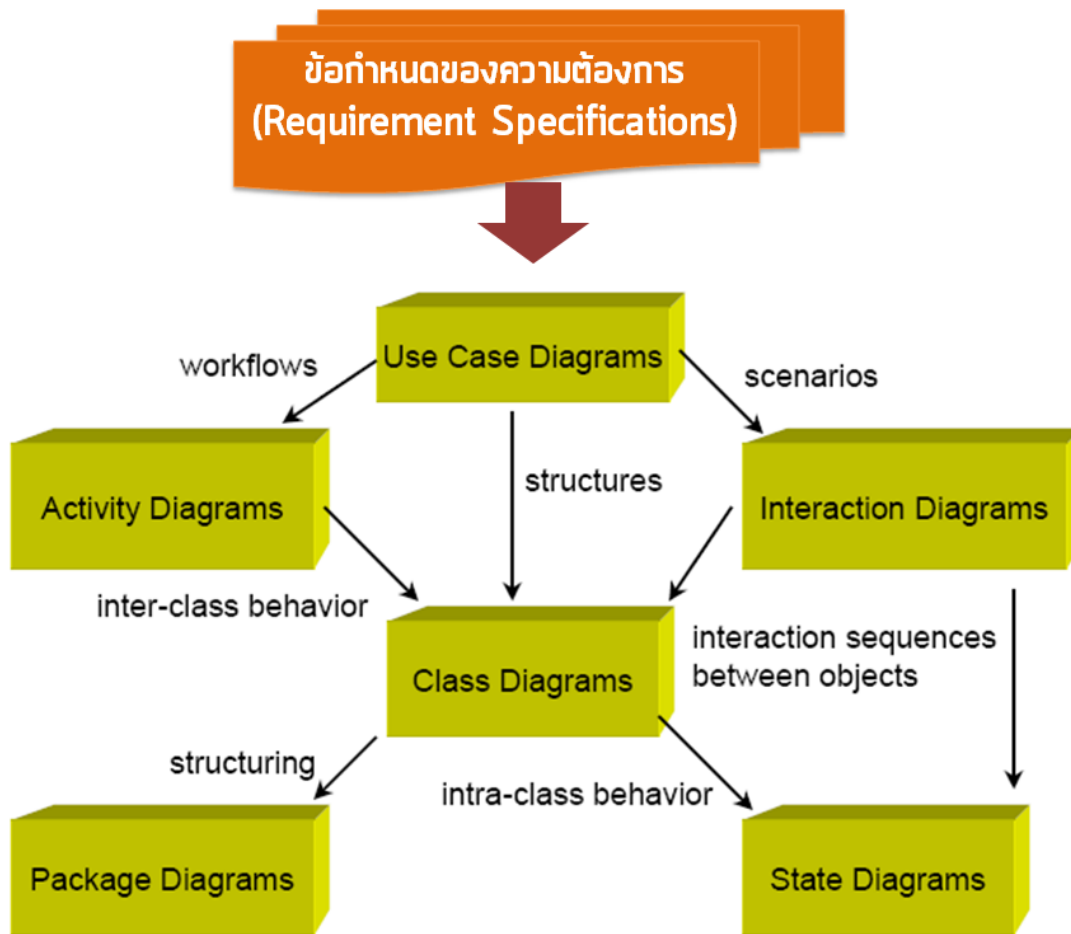
(ที่มา : <https://ritablueraayitsolutions.wordpress.com/2014/03/01/154/>)

2) เทคนิคการทำต้นแบบ(Prototyping)

เทคนิคการทำต้นแบบ (Prototyping) จะใช้ในกรณีที่ผู้ใช้อย่างไม่ทราบความต้องการของตนเองอย่างชัดเจน โดยสร้างต้นแบบของระบบอย่างรวดเร็วจากความต้องการที่มีอยู่เบื้องต้น จากนั้นให้ผู้ใช้ประเมินและเสนอแนะเพื่อทำการปรับปรุงจนกว่าผู้ใช้อยอมรับ เสร็จแล้วนำต้นแบบที่เป็นที่ยอมรับของผู้ใช้มาพัฒนาต่อให้สมบูรณ์จนสามารถใช้งานได้จริง



ภาพที่ 5.3 ตัวอย่างการทำต้นแบบ



ภาพที่ 5.4 แสดงทิศทางของแผนภาพต่าง ๆ ในการพัฒนาระบบ

การนำเสนอความต้องการด้วยแผนภาพยูสเคส

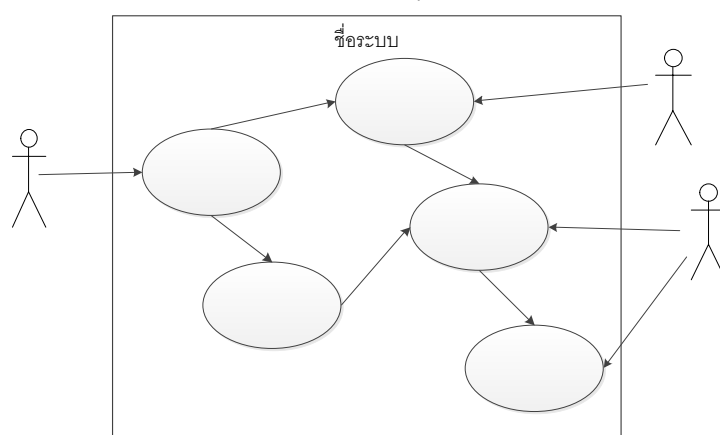
ความต้องการของระบบ เป็นสิ่งแรกที่ผู้พัฒนาระบบต้องคำนึงถึงในการพัฒนาระบบงาน โดยนักพัฒนาระบบจะต้องจัดบันทึกสิ่งต่าง ๆ ให้ครบถ้วนถูกต้องและชัดเจน โดยใช้เทคนิคต่าง ๆ ในการเก็บรวบรวมข้อมูลความต้องการ (Information Gathering) ซึ่งเป็นการค้นหว่าระบบงานต้องการอะไร เพื่อเป็นการยืนยันความเข้าใจที่ตรงกันระหว่างผู้พัฒนาและผู้ใช้งาน (พนิดา พานิชกุล, 2548) อย่างไรก็ตามการได้ความต้องการของระบบที่ครบถ้วนและสมบูรณ์ในขั้นแรกเป็นเรื่องยาก เนื่องจากความต้องการของระบบอาจมีการเปลี่ยนแปลงหรือเพิ่มเติมได้ในภายหลังจากที่มีการพัฒนาระบบงานไปแล้ว ในปัจจุบันจึงมีเครื่องมือในการออกแบบระบบงานโดยการสร้างแบบจำลองระบบที่สามารถจำลองระบบย่อยต่าง ๆ ได้ โดยแต่ละระบบมีการสร้างความสัมพันธ์กับผู้ใช้งาน และยังสามารถอธิบายปฏิสัมพันธ์ระหว่างระบบงานกับสิ่งที่อยู่ภายนอกระบบได้ ซึ่งเครื่องมือดังกล่าวนี้ เรียกว่า แบบจำลองยูสเคส (Use Case Model) (ธนุวิส อัสรานนทกุล, 2555)

แบบจำลองยูสเคสถูกพัฒนาขึ้นโดย Ivar Jacobson ในปี ค.ศ. 1986 โดยเริ่มสร้างโครงสร้างและแบบจำลองทางเทคนิคสำหรับกำหนดยูสเคส (Use Case) เป็นชุดของกิจกรรมเพื่อสร้างผลลัพธ์ออกมา แต่ละยูสเคส จะอธิบายว่าระบบมีการตอบสนองต่อเหตุการณ์ที่เกิดขึ้นอย่างไร เพื่อนำมาใช้ในการพัฒนาปรับปรุง

เทคนิค หน้า ที่ ความต้องการของระบบให้ดีขึ้น และสามารถนำเข้ามาใช้ได้ และได้เขียนหนังสือ เรื่อง Objected-Oriented Software Engineering A Use Case Driven Approach หนังสือเล่มนี้ถูกเขียนขึ้น จากประสบการณ์การทำงานของผู้เขียน ในการพัฒนาระบบงาน ให้กับบริษัทต่าง ๆ โดยใช้ ยูสเคสเป็นหลักในการพัฒนาระบบงาน ทำให้เป็นการยืนยันได้ว่าการใช้แบบจำลองยูสเคสเป็นสิ่งสำคัญในการวางระบบเพื่อพัฒนาระบบงาน (กิติ ภัคตีวัฒน์กุล, และกิตติพงษ์ กลมกล่อม, 2544)

ในบทที่แล้วได้อธิบายเกี่ยวกับยูเอ็มแอลและแผนภาพแบบต่าง ๆ ไปบ้างแล้วโดยเฉพาะแผนภาพ ยูสเคส ซึ่งคือแบบจำลองความต้องการของระบบที่นำเสนอความต้องการเชิงฟังก์ชัน (Functional Requirement) ของระบบโดยรวมจากมุมมองของผู้ใช้ภายนอกหรือระบบภายนอก โดยระบุพฤติกรรมหรือ หน้าที่การทำงานของระบบที่เชื่อว่าอะไรที่ระบบจำเป็นต้องมี โดยยูสเคสจะใช้ในการทดสอบและตรวจสอบ โครงสร้างและหน้าที่การทำงานของระบบซึ่งจะนำเสนอยูสเคสและการปฏิสัมพันธ์โต้ตอบกันระหว่างระบบ และผู้ใช้ภายนอกซึ่งอาจเป็นคนหรือระบบก็ได้ในยูเอ็มแอลนั้นยูสเคสจะสามารถเขียนได้ 2 ลักษณะ ดังนี้คือ 1. คำอธิบายยูสเคส (Use Case Description) และ 2. ยูสเคสแบบแผนภาพ (Use Case Diagram) (กิตติพงษ์ กลมกล่อม, 2552).

1. แผนภาพยูสเคส (Use Case Diagram) เป็นแผนภาพที่แสดงขอบเขตของงาน หน้าที่ในการทำงานและ ผู้เกี่ยวข้องภายนอก และใช้ในระบบงานสำหรับสื่อสารระหว่างผู้พัฒนาระบบกับผู้ใช้งานเป็นวิธีการค้นหาหรือ เก็บความต้องการที่แท้จริงของผู้ใช้ร่วมกับเอกสารขอบเขตของระบบที่จะพัฒนา (Problem Domain) เพื่อให้ ได้ความต้องการที่แท้จริง จากภาพที่ 5.5 แสดงตัวอย่างแผนภาพยูสเคส ที่ประกอบด้วยแอ็คเตอร์และยูสเคสที่ อยู่ภายในระบบงาน สัญลักษณ์ยูสเคสจะแทนด้วยวงรี แอ็คเตอร์จะแทนด้วยรูปกระดุกคน โดยที่แผนภาพยูส เคส จะแสดงถึงความสัมพันธ์ระหว่างยูสเคสและมีการติดต่อกับแอ็คเตอร์ หรือ แผนภาพยูสเคสจะเป็นการ อธิบายถึงลำดับของเหตุการณ์ต่างๆ ความสัมพันธ์ของเหตุการณ์ภายในที่เกิดขึ้นจากแอ็คเตอร์ และนำผลที่ได้ จากการกระทำไปใช้งานต่อ (Object Management Group, 1997)



ภาพที่ 5.5 แผนภาพยูสเคส

2. คำอธิบายยูสเคส (Use Case Description) เป็นการเขียนเอกสารรายละเอียด หรือคำบรรยายสำหรับ ยูสเคส ในรูปแบบของการบรรยายเหตุการณ์หลักจะอธิบายถึงชุดเหตุการณ์หลักที่เกิดขึ้นในการปฏิบัติงานของ

ยูสเคสหนึ่ง ๆ ของระบบตามลำดับ จะเป็นการช่วยให้การทำงานหรือการสร้างระบบงานใหม่นั้นประสบความสำเร็จได้ง่ายขึ้น และเพื่อนำไปใช้ในการวิเคราะห์ สร้างแม่แบบทั่วไปที่ใช้ในการเขียนคำอธิบายยูสเคส ซึ่งจะกล่าวในหัวข้อต่อถัด ๆ ไป

วัตถุประสงค์ของการใช้งานแผนภาพยูสเคส

ในการใช้งานแผนภาพยูสเคสสำหรับการวิเคราะห์ความต้องการมีวัตถุประสงค์ ดังนี้

- 1) เพื่ออธิบายเรื่องราวของขอบเขตของระบบทั้งหมด
- 2) บอกส่วนประกอบในระบบว่าระบบประกอบด้วยระบบย่อยอะไรบ้าง
- 3) บอกความสัมพันธ์ของส่วนต่าง ๆ ในระบบ

ประโยชน์ของแผนภาพยูสเคส


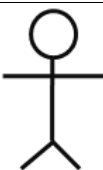
- 1) ช่วยให้ผู้พัฒนาระบบสามารถแยกแยะกิจกรรมที่อาจเกิดขึ้นในระบบ
- 2) เป็นแผนภาพพื้นฐานที่สามารถอธิบายสิ่งต่าง ๆ ได้โดยใช้รูปภาพที่ไม่ซับซ้อน
- 3) แผนภาพยูสเคสจะมีประสิทธิภาพ หากผู้เขียนมีความเข้าใจในขอบเขตของระบบอย่างแท้จริง

โดยส่วนประกอบของแผนภาพยูสเคสจะประกอบด้วย

- 1) ยูสเคส (Use Case) หมายถึง ความสามารถหรือหน้าที่ของระบบ
- 2) แอกเตอร์ (Actor) หมายถึง ผู้กระทำหรือผู้ใช้งานยูสเคสนั้น ๆ
- 3) ความสัมพันธ์ (Relationship) หมายถึง เส้นแสดงความสัมพันธ์ระหว่างยูสเคสและแอกเตอร์
- 4) ระบบ (System) หมายถึง ระบบที่กำลังพัฒนา

โดยมีรายละเอียดดังแสดงในตารางที่ 5.1

ตารางที่ 5.1 แสดงสัญลักษณ์ต่าง ๆ ที่ใช้ในแผนภาพยูสเคส

สัญลักษณ์	ชื่อ	ความหมาย/หน้าที่
	ยูสเคส (Use case)	หมายถึง ความสามารถหรือหน้าที่ของระบบที่ตอบสนองกับแอกเตอร์หรือมีปฏิสัมพันธ์กับแอกเตอร์
	แอกเตอร์ (Actor)	หมายถึง ผู้เกี่ยวข้องกับระบบซึ่งเป็นผู้กระทำหรือผู้ใช้งานยูสเคสนั้น ๆ หรือทำหน้าที่ผลักดันให้เกิดกิจกรรมของระบบหรือทำหน้าที่ควบคุมดูแลกิจกรรมของระบบหรือสัมผัสกับระบบโดยตรง ซึ่งอาจเป็นบุคคล กลุ่มบุคคล หรือสิ่งของก็ได้

_____	ความสัมพันธ์ (Relationship)	หมายถึงเส้นแสดงความสัมพันธ์ระหว่าง ยูสเคสและแอกเตอร์
□	ระบบ (System)	หมายถึงระบบที่กำลังพัฒนา

1. ยูสเคสและสถานการณ์สมมุติ

1) ยูสเคส หมายถึงความสามารถหรือหน้าที่การทำงานของระบบ โดยแต่ละยูสเคสแทนชุดของรายการคำสั่ง (Transactions) ที่ระบบทำงานได้ต่อกับผู้ใช้งานหรือระบบอื่น ๆ ภายนอก

2) สถานการณ์สมมุติ (Scenario) หมายถึงสถานการณ์หรือตัวอย่างเรื่องราวการใช้งานระบบ สถานการณ์สมมุติจัดเป็นอินสแตนซ์ของยูสเคส เช่น นายรัฐพงศ์ กดเงินจากตู้เอทีเอ็มของธนาคารกสิกร จำนวน 2,000 บาท

2. แอกเตอร์

แอกเตอร์ (Actors) หมายถึงคนหรือสิ่งต่าง ๆ ที่มีการปฏิสัมพันธ์ได้ต่อกับระบบ หรือสิ่งใดก็ตามที่มีความต้องการในการแลกเปลี่ยนข้อมูลข่าวสารกับระบบหรือสิ่งใดที่อยู่ภายนอกระบบ และมีการใช้งานยูสเคสของระบบสิ่งที่กำหนดบทบาทหน้าที่ของผู้ใช้ระบบและสิ่งที่กำหนดการเชื่อมโยงกับระบบภายนอก ดังตัวอย่างที่ 5.3

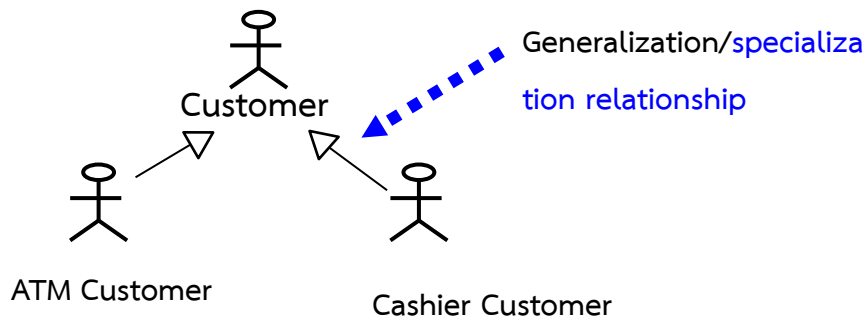
ตัวอย่างที่ 5.3 ตัวอย่างของแอกเตอร์

1. Customer -- maintain their account
2. Cashier -- verify withdrawal amount



ภาพที่ 5.6 แสดงตัวอย่างของแอกเตอร์

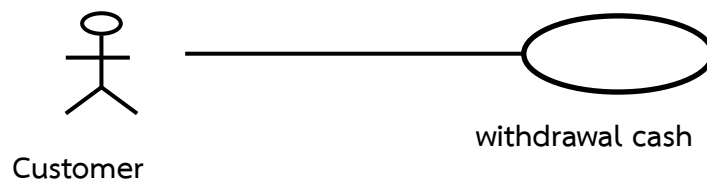
นอกจากนี้ความสัมพันธ์ระหว่างแอกเตอร์นั้นสามารถใช้ความสัมพันธ์เชิงเงื่อนไขไลเซนซ์อธิบายได้ดังแสดงในภาพที่ 5.7



ภาพที่ 5.7 แสดงตัวอย่างของความสัมพัทธ์ระหว่างแอกเตอร์

การใช้ความสัมพันธ์แบบเจเนอรัลไลเซชัน (Generalization Relationships) อธิบายความสัมพันธ์ระหว่างแอกเตอร์อาจไม่จำเป็นต้องอธิบายรายละเอียดของแอสโซซิเอชันเนื่องจากไม่มีการนำไปใช้ในขั้นตอนการเขียนโปรแกรมสำหรับแอกเตอร์ในระบบ

ในบางครั้งอาจพิจารณาแอกเตอร์เป็นเหมือนคลาสในยูเอ็มแอลเนื่องจากสามารถมีความสัมพันธ์ระหว่างกันเช่นเดียวกับที่คลาสมิซึ่งแอกเตอร์จะเชื่อมต่อกับยูสเคส โดยเส้นตรงจะแสดงความเกี่ยวข้องหรือปฏิสัมพันธ์ที่ความสัมพันธ์ของการติดต่อสื่อสารกันทั้งการรับและส่งข่าวสารให้แก่กันและกันดังแสดงในภาพที่ 5.8



ภาพที่ 5.8 ตัวอย่างแอกเตอร์และยูสเคสของการถอนเงินสด

3. ระบบ(System)

ความหมายของระบบ(System)

ก่อนที่จะทำการวิเคราะห์ระบบนั้นควรทำความเข้าใจและทำความรู้จักกับระบบก่อนว่าระบบคืออะไร หมายถึงอะไรมีส่วนประกอบหรือองค์ประกอบที่จะประกอบเป็นระบบได้อย่างไรซึ่งได้มีผู้ให้คำจำกัดความและความหมายของระบบเอาไว้หลายความหมายด้วยกันดังนี้

ระบบ (System) มีความหมายตามพจนานุกรม ฉบับราชบัณฑิตยสถานพ.ศ. 2525 ได้ให้ความหมายเอาไว้ว่า ระบบคือระเบียบเกี่ยวกับการรวมสิ่งต่าง ๆ ซึ่งมีลักษณะซับซ้อนให้เข้าลำดับประสานเป็นอันเดียวกันตามหลักเหตุผลทางวิชาการหรือหมายถึงปรากฏการณ์ทางธรรมชาติซึ่งมีความสัมพันธ์ประสานเข้ากันโดยกำหนดรวมเป็นอันหนึ่งอันเดียวกัน

ระบบ (System) คือ กระบวนการต่าง ๆ ที่อยู่ในเครือข่ายเดียวกันและมีความสัมพันธ์กันระหว่างกระบวนการเหล่านั้นและเชื่อมต่อกันเพื่อทำงานใดงานหนึ่งให้บรรลุถึงเป้าหมายที่วางไว้

ระบบ (System) คือ กลุ่มขององค์ประกอบต่าง ๆ ที่ทำงานร่วมกัน เพื่อจุดประสงค์อันเดียวกัน และเพื่อให้เข้าใจในความหมายของคำว่าระบบที่จะต้องทำการวิเคราะห์จึงต้องเข้าใจลักษณะของระบบก่อน (ที่มา : <http://gangza-yui.blogspot.com/p/system.html>)

ความหมายของระบบ (System) ในกระบวนการวิเคราะห์ความต้องการด้วยยูสเคสหมายถึงทั้งระบบที่เป็นฮาร์ดแวร์และซอฟต์แวร์ โดยวัตถุประสงค์ในแผนภาพยูสเคสเพื่อจะระบุขอบเขตของระบบที่กำลังพัฒนา(System Boundary) ซึ่งจะใช้สัญลักษณ์สี่เหลี่ยมแทนระบบดังแสดงในภาพที่ 5.9


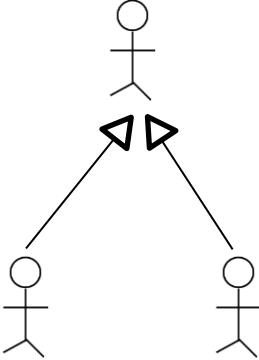

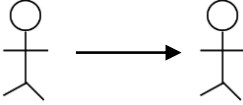


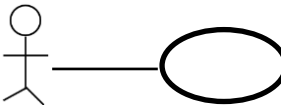




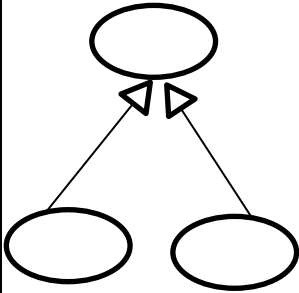
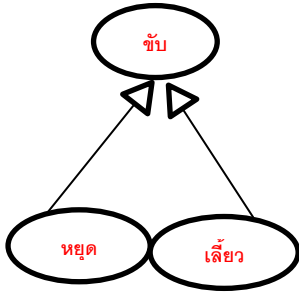
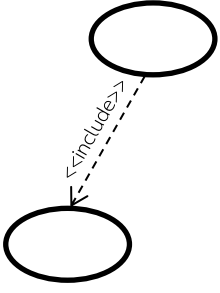
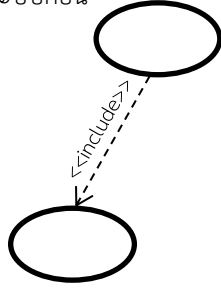
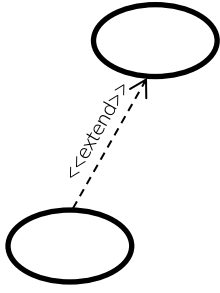
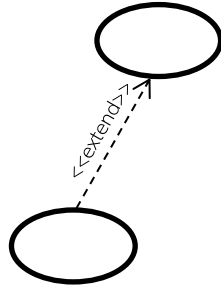
ภาพที่ 5.9 แสดงสัญลักษณ์ของระบบ (System)

ประเภทของความสัมพันธ์

กิจกรรมที่เกิดขึ้นภายในแบบจำลองยูสเคส อาจทำให้เกิดความสัมพันธ์ในรูปแบบต่าง ๆ ได้ ความสัมพันธ์ระหว่างแอกเตอร์กับแอกเตอร์ ความสัมพันธ์ระหว่างแอกเตอร์กับยูสเคส และความสัมพันธ์ระหว่างยูสเคสกับยูสเคสโดยมีรายละเอียด ดังตารางที่ 5.2

ตารางที่ 5.2 แสดงประเภทความสัมพันธ์

ประเภทของความสัมพันธ์	1.1 ความสัมพันธ์แบบทั่วไป (Generalization Relation)	ความหมาย	ตัวอย่าง
<p>1. ระหว่างแอกเตอร์กับแอกเตอร์</p> 	<p>1.1 ความสัมพันธ์แบบทั่วไป (Generalization Relation)</p> 	<p>ความสัมพันธ์แบบทั่วไป เป็นความสัมพันธ์แบบถ่ายทอดหรือรับทอด (inheritance) เป็นความสัมพันธ์ที่แอกเตอร์ลูกรับทอดคุณสมบัติหรือฟังก์ชันต่าง ๆ จากแอกเตอร์พ่อแม่โดยการรวมคุณสมบัติบางประการเหมือนหรือคล้ายกันจัดให้เป็นแอกเตอร์กลุ่มเดียวกัน</p> <p>สัญลักษณ์ที่ใช้</p> 	<ul style="list-style-type: none"> - ลูกค้าของร้านขายสินค้าออนไลน์แบ่งเป็น 2 ประเภทคือลูกค้าทั่วไปกับลูกค้าประจำ - ลูกค้าของธนาคารสามารถแบ่งออกเป็น 2 ประเภทคือลูกค้าส่วนบุคคล (Personal Customer) และลูกค้าระดับองค์กร (Cooperate Customer) - ผู้ใช้งานห้องสมุด แบ่งเป็น อาจารย์ และ นักเรียน
	<p>1.2 ความสัมพันธ์แบบแอสโซซิเอชัน (Association)</p> 	<p>ความสัมพันธ์แบบแอสโซซิเอชัน การที่แอกเตอร์กระทำร่วมกันหรือผ่านไปยังอีกแอกเตอร์หนึ่ง</p> <p>สัญลักษณ์ที่ใช้</p> 	<ul style="list-style-type: none"> - ลูกค้าโรงแรมต้องการจองห้องพักของโรงแรมโดยการแจ้งผ่านไปยังพนักงานของโรงแรมเพื่อทำการตรวจสอบห้องว่าง และ จองห้องพักโรงแรมให้ลูกค้า
<p>2. ระหว่างแอกเตอร์กับยูสเคส</p> 	<p>2.1 ความสัมพันธ์เป็นแบบแอสโซซิเอชัน</p> 	<p>การแอกเตอร์จะกระทำกับยูสเคสในลักษณะการเรียกใช้งานฟังก์ชันของระบบหรือการมีความเกี่ยวข้องกับยูสเคส เรียกว่าความสัมพันธ์เป็นแบบแอสโซซิเอชัน</p> <p>สัญลักษณ์ที่ใช้</p> 	<ul style="list-style-type: none"> - ลูกค้าสมัครสมาชิกใหม่ - ลูกค้าซื้อสินค้า - ลูกค้าจองห้องพัก - พนักงานบันทึกข้อมูลสินค้า 

			- พนักงานจองห้องพัก
3. ระหว่างยูสเคสกับยูสเคส	3.1 ความสัมพันธ์แบบทั่วไป	 <p>เป็นความสัมพันธ์แบบถ่ายทอดหรือรับทอด (inheritance) เป็นความสัมพันธ์ที่ยูสเคสลูกรับทอดคุณสมบัติหรือฟังก์ชันต่าง ๆ จากยูสเคสพ่อแม่โดยการรวมคุณสมบัติบางประการเหมือนหรือคล้ายกันจัดให้เป็นยูสเคสกลุ่มเดียวกัน</p>	 <p>การขับรถ</p>
	3.2 ความสัมพันธ์แบบรวม	 <p>เป็นความสัมพันธ์ที่นำขั้นตอนการทำงานที่ซ้ำซ้อนมาสร้างเป็นยูสเคสแยกต่างหากเพื่อให้ยูสเคสอื่นเรียกใช้ได้</p>	<p>- ก่อนการจองตั๋ว ต้องเข้าระบบก่อน</p> 
	3.3 ความสัมพันธ์แบบขยาย	 <p>เป็นความสัมพันธ์ที่มีการดำเนินกิจกรรมของตนเองอย่างตามปกติ แต่อาจมีสิ่งกระตุ้นแล้วทำให้เกิดการทำงานที่แตกต่างออกไป</p>	<p>- การชำระค่าปรับเมื่อเลยกำหนดคืนหนังสือที่เช่าไป</p> 

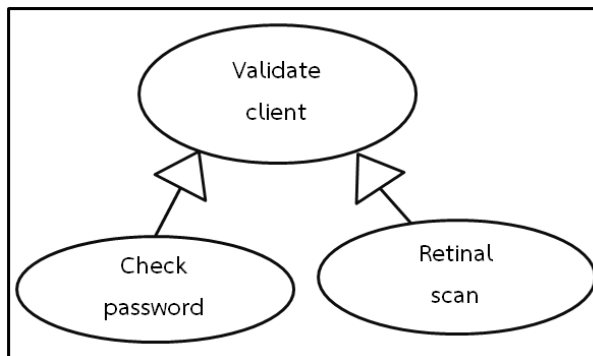
ความสัมพันธ์ระหว่างยูสเคสกับยูสเคส

ความสัมพันธ์ระหว่างยูสเคสกับยูสเคส (Relationships Between Use Case) หมายถึงในแต่ละยูสเคสจะสามารถมีความสัมพันธ์ระหว่างกันได้ 2 แบบคือ Extend และ Uses หรือ Include โดย

- 1) Extends เป็น Generalization Relationships ในกรณีที่ยูสเคสหนึ่งๆ ขยายความสามารถ (extends) ของยูสเคสอื่น ๆ โดยการเพิ่มการกระทำ (Actions)
- 2) Includes หรือ Uses เป็นความสัมพันธ์แบบเจเนอรัลไลเซชันในกรณีที่ยูสเคสหนึ่ง ๆ เรียกใช้ยูสเคสอื่นที่พิจารณาให้เป็นส่วนหนึ่งของยูสเคสนั้น ๆ

ความสัมพันธ์แบบเจเนอรัลไลเซชัน

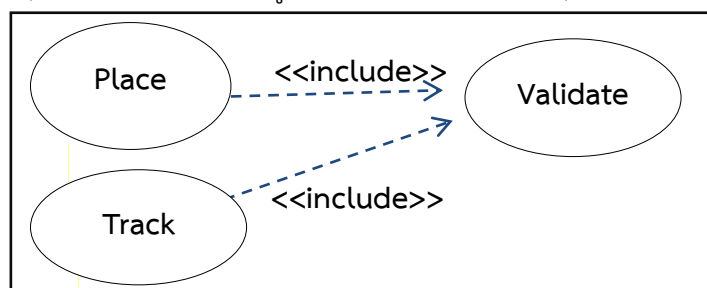
ความสัมพันธ์แบบเจเนอรัลไลเซชัน (Generalization relationship) หมายถึง Child use case รับถ่ายทอดคุณสมบัติมาจาก Parent Use Case ซึ่ง Child สามารถเปลี่ยนแปลงพฤติกรรมที่รับจาก Parent หรือเพิ่มเติมพฤติกรรม และ Child อาจนำไปแทนที่ในที่ ๆ Parent ปรากฏดังแสดงในภาพที่ 5.10



ภาพที่ 5.10 แสดงความสัมพันธ์ระหว่างยูสเคสแบบเจเนอรัลไลเซชัน

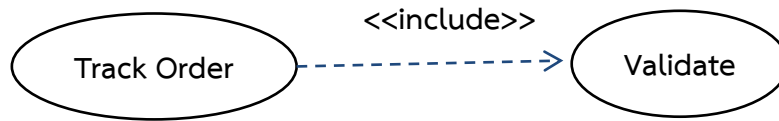
ความสัมพันธ์แบบรวม

ความสัมพันธ์แบบรวม (Include หรือ Use) โดยมากมักจะใช้ในการหลีกเลี่ยงการอธิบายการไหลของเหตุการณ์ (Flow of Events) เดิมซ้ำกันหลาย ๆ ครั้ง โดยรวมพฤติกรรมรวมในยูสเคสเพื่อหลีกเลี่ยงการคัดลอกและวาง (copy & paste) ของคำอธิบายยูสเคส (Use Case Descriptions)



ภาพที่ 5.11 แสดงตัวอย่างของความสัมพันธ์แบบ Include Relationship

ตัวอย่างที่ 5.4 ตัวอย่างการใช้งานยูสเคสที่มี “Include” ประกอบ



ภาพที่ 5.12 ตัวอย่างการใช้งานยูสเคสที่มี “Include” ประกอบ

การเขียนคำอธิบายยูสเคส (Use Case Description)

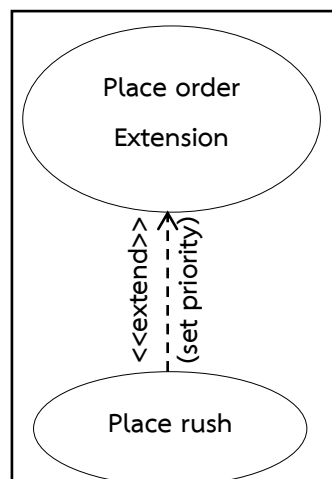
Name : การตรวจสอบรายการซื้อขายหลักทรัพย์ (Track Order)

Main flow :

1. ใช้หมายเลข order ในการตรวจสอบที่ได้รับจากตลาดหลักทรัพย์ Obtain and verify order number
2. *Include* ส่วนของ “Validate client”
3. ในแต่ละส่วนของ Order ...

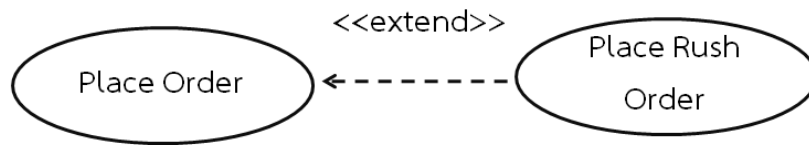
ความสัมพันธ์แบบขยาย

ความสัมพันธ์แบบ (Extend) จะใช้สร้างแบบจำลองบางส่วนของยูสเคสที่ผู้ใช้อาจมองว่าจำเป็นหรือไม่จำเป็นต้องมี (Optional) ก็ได้ หรืออาจใช้สำหรับสร้างแบบจำลอง Conditional Sub Flows และใช้ในการแทรก Sub Flows ในจุดที่ระบุโดยพิจารณาปฏิสัมพันธ์ระหว่างแอกเตอร์



ภาพที่ 5.13 ตัวอย่างการใช้งานยูสเคสที่มี “Extend” ประกอบ

ตัวอย่างที่ 5.5 การใช้งานยูสเคสที่มี “Extend” ประกอบของยูสเคสที่มีการวางคำสั่งซื้อแบบเร่งด่วน



ภาพที่ 5.14 แสดงตัวอย่างการใช้ extend

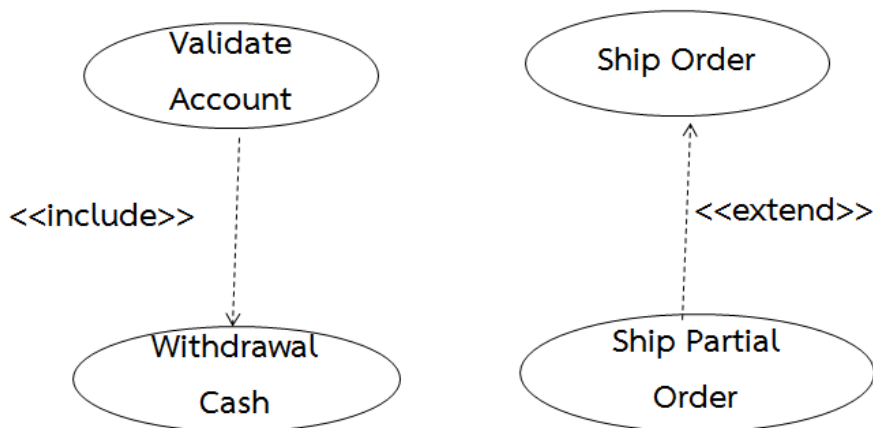
คำอธิบายยูสเคส

Name: การส่งรายการซื้อขายหลักทรัพย์ (Place order)

Main flow of events:

1. ...
2. ผู้ซื้อ (Trader) ป้อนเงื่อนไขของหลักทรัพย์ ที่เครื่องลูกข่าย (Client) ต้องการซื้อขาย
3. กำหนดลำดับความสำคัญโดย (*Set priority*)
4. ระบบส่งคำสั่งซื้อ (Order) ให้กับตลาดหลักทรัพย์
5. ...

ความสัมพันธ์ระหว่างยูสเคส (Relationships between use case)



ภาพที่ 5.15 เปรียบเทียบระหว่าง extends กับ include

ตารางที่ 5.3 เปรียบเทียบความสัมพันธ์ระหว่างแบบ extends กับ uses หรือ include

uses/include	extends
1. ใช้ extract พฤติกรรมร่วม 2. มักไม่มีแอกเตอร์เกี่ยวข้องโดยตรงกับยูสเคสที่มีพฤติกรรมร่วม 3. แอกเตอร์ที่ต่างกัน for “caller” use cases possible	1. ใช้แยกความแตกต่างของยูสเคส 2. แอกเตอร์ที่เกี่ยวข้องมักเป็นคนกระทำยูสเคสและยูสเคสที่ extend ทั้งหมด 3. แอกเตอร์มักเชื่อมต่อกับ “base” ยูสเคส

การใช้งานยูสเคส

ในการพัฒนาระบบด้วยหลักการเชิงวัตถุเมื่อต้องการหาความต้องการ (Requirements capture) นักวิเคราะห์ระบบจะใช้ในการกำหนดความต้องการของระบบจากนั้นสร้างมาเป็นแบบจำลองของความต้องการของผู้ใช้ด้วยยูสเคสไดอะแกรม และใช้ในการทดสอบสถานการณ์สมมุติ (Scenarios testing) โดยสร้างเป็นแบบจำลองของสถานการณ์การทดสอบระบบด้วยยูสเคส เช่นกัน (Shin, S., 2015) โดยมีขั้นตอนการวิเคราะห์ความต้องการโดยใช้ยูสเคส ดังนี้

1. การระบุหายูสเคส

วิธีการในการระบุหายูสเคสสามารถทำได้โดยระบุสิ่งที่ลูกค้าหรือผู้ใช้งานต้องการให้มีในระบบแล้วตั้งชื่อให้กับยูสเคสโดยเขียนเป็นคำอธิบายสั้นๆ แล้วค่อยเพิ่มเติมรายละเอียดต่าง ๆ ในภายหลังเพื่อให้ได้แผนภาพที่ครบถ้วนสมบูรณ์

2. การค้นหาแอกเตอร์

ในการวิเคราะห์ระบบโดยใช้ยูสเคสนั้นสามารถค้นหาแอกเตอร์โดยระบุหรือหาแอกเตอร์ได้โดยตอบคำถามต่อไปนี้

- 1) ใครเป็นคนใช้งานหน้าที่การทำงานหลักของระบบ
- 2) ใครต้องการการสนับสนุนการทำงานจากระบบ
- 3) ใครต้องการบำรุงรักษาและบริหารระบบ
- 4) อุปกรณ์ใดที่ต้องการให้ระบบจัดการดูแล
- 5) ระบบภายนอกระบบใดที่ต้องการให้ระบบมีปฏิสัมพันธ์ด้วย
- 6) ใครหรืออะไรที่ต้องการได้รับผลประโยชน์จากเข้าที่พู่ของระบบ

เทคนิคเพิ่มเติม (Tips) ไม่ควรพิจารณาเฉพาะผู้ใช้งานระบบที่ใช้จากระบบโดยตรงแต่ให้พิจารณาผู้ใช้งานอื่นๆ ที่ต้องการใช้บริการจากระบบด้วย

สำหรับแต่ละแอกเตอร์ตอบคำถามต่อไปนี้

- 1) หน้าที่การทำงานอะไรที่แอกเตอร์ต้องการจากระบบ
- 2) ข้อมูลใดบ้างที่แอกเตอร์ต้องการสร้างอ่านลบเปลี่ยนแปลงหรือเก็บอยู่ภายในระบบ
- 3) เหตุการณ์ใดบ้างที่ระบบต้องแจ้งให้แอกเตอร์ทราบหรือแอกเตอร์ต้องแจ้งให้ระบบทราบ
- 4) หน้าที่การทำงานของระบบช่วยทำให้งานประจำวันของแอกเตอร์ง่ายขึ้นหรือไม่

นอกจากนี้ถ้าไม่พิจารณาที่ตัวแอกเตอร์ ก็ลองพิจารณาสິงเหล่านี้เพิ่มเติม

- 1) อะไรคือสิ่งนำเข้าหรือออกของระบบและอินพุตหรือเอาต์พุตเหล่านั้นมาจากไหนหรือใครเป็นคนนำไปใช้งาน
- 2) ปัญหาหลักของระบบที่ใช้งานอยู่คืออะไร

สูตรหรือเทคนิคในการค้นหาแอกเตอร์

- 1) ระบุแอกเตอร์ที่มีปฏิสัมพันธ์กับระบบ
- 2) พิจารณาแนวทางของระบบในการปฏิสัมพันธ์กับแอกเตอร์
- 3) จำแนกพฤติกรรมของระบบในการปฏิสัมพันธ์กับแอกเตอร์ให้เป็นยูสเคสโดยกำหนดความสัมพันธ์ระหว่างยูสเคส

แนวทางการสร้างยูสเคส

การสร้างยูสเคสเพื่ออธิบายความต้องการมีสิ่งที่จะต้องพิจารณา ดังนี้

1) การจำแนกแอกเตอร์

การจำแนกแอกเตอร์ จะเป็นการหาว่าใครเป็น แอกเตอร์ หรือ คน สิ่งของ เพื่อให้ผู้วิเคราะห์สามารถบอกได้ว่า ระบบที่สร้างนี้ใครหรือสิ่งใดจะเป็นผู้ใช้ระบบ โดยที่ยังไม่ต้องคิดว่าจะทำได้อย่างไร เมื่อมีการวิเคราะห์และทำการจำแนกแอกเตอร์ได้ และสามารถบอกขอบเขตของระบบงานที่สนใจอยู่ว่าจะได้ผลลัพธ์บางอย่างออกมาเพื่อให้แอกเตอร์รับไปใช้งานต่อไป วิธีที่ช่วยทำให้สามารถหาว่าอะไรคือผู้ที่กระทำในระบบ สามารถวิเคราะห์ได้ดังนี้

1. ผู้ใด ใครหรือสิ่งใด เป็นผู้ที่นำข้อมูลเข้าสู่ระบบ
2. ผู้ใด ใครหรือสิ่งใด เป็นผู้ที่แสดงผลข้อมูลที่ได้จากระบบ
3. ผู้ใด ใครหรือสิ่งใด มีปฏิสัมพันธ์กับระบบอื่นๆ

2) การจำแนกยูสเคส

การจำแนกยูสเคสตามความต้องการของระบบ เป็นการพิจารณาว่าระบบต้องการอะไรเพื่อให้ได้ยูสเคส ในบางระบบอาจจะมียูสเคสจำนวนมาก ซึ่งยูสเคสควรจะเป็นงานที่มีความสำคัญ มีความซับซ้อน และจำเป็นที่จะต้อง

มี ดังนั้นในการพิจารณาแอ็คเตอร์ หรือผู้กระทำจะต้องมีการปฏิสัมพันธ์กับยูสเคสเท่านั้น สามารถวิเคราะห์ได้ ดังนี้

1. งานหลักของแอ็คเตอร์คืออะไร
2. แอ็คเตอร์ มีงานสารสนเทศใดบ้าง
3. ข้อมูลหรือสารสนเทศใดบ้างที่แอ็คเตอร์จำเป็นต้องมีการป้อนเข้าสู่ระบบ
4. ควรมีการระบุภายในระบบว่ามีความจำเป็นหรือไม่ที่จะต้องมีการแจ้งแอ็คเตอร์ถ้าเกิดมีการเปลี่ยนแปลงขึ้น และในทางกลับกันแอ็คเตอร์นั้นจำเป็นหรือไม่ที่จะต้องมีการแจ้งต่อระบบ เมื่อมีการเปลี่ยนแปลงหรือมีเหตุการณ์เกิดขึ้น เป็นต้น

3) การสร้างแผนภาพยูสเคส

การสร้างแผนภาพยูสเคสจะถูกสร้างทันทีเมื่อได้แอ็คเตอร์และสามารถระบุขอบเขตของระบบได้แล้ว เป็นการสร้างความสัมพันธ์ภายในระบบ แต่ถ้ามีภายในระบบเกิดมียูสเคสจำนวนมากควรจะมีการแบ่งแยกยูสเคสเป็นระบบย่อยๆ เพื่อสะดวกแก่การอ่านและเข้าใจในการทำงานของระบบมากขึ้น

4) การจัดทำเอกสารคำอธิบายยูสเคส

การจัดทำเอกสารคำอธิบายยูสเคส จะเกิดขึ้นหลังจากที่มีการสร้างแผนภาพยูสเคสของธุรกิจนั้นเสร็จเรียบร้อยแล้ว เพื่อเป็นการอธิบายแผนภาพ และระบุฟังก์ชันต่างๆให้มีความชัดเจนมากขึ้น และสร้างความมั่นใจแก่ผู้ที่เขียนว่าได้สร้างความสัมพันธ์ยูสเคสได้อย่างถูกต้องตามความต้องการจริง ดังนั้นแผนภาพที่ถูกสร้างขึ้นมานี้สามารถมาใช้ในการออกแบบและพัฒนาระบบ รวมถึงใช้ในขั้นตอนการทดสอบระบบได้ จะกล่าวโดยละเอียดในบทต่อ ๆ ไป

5) การสร้างความสัมพันธ์ระหว่างยูสเคส

การสร้างความสัมพันธ์ระหว่างยูสเคส เป็นการระบุความสัมพันธ์ระหว่างยูสเคส เพื่อช่วยในเรื่องของการเชื่อมต่อตามความสัมพันธ์หรือการเชื่อมโยงของยูสเคสที่เกี่ยวข้องกัน ทำให้สามารถเข้าใจภาพรวมของระบบงานภายในได้ทั้งหมด

การเขียนยูสเคสเป็นการนำสิ่งที่ผู้ใช้งานจริงต้องการออกมาเป็นแบบจำลอง ดังนั้นหลักการของการเขียนยูสเคส คือ จะต้องทำให้ง่ายที่สุดเท่าที่จะทำได้ เพราะผู้ใช้ระบบงานมีความรู้ทางด้านคอมพิวเตอร์น้อย ผลของหลักการข้อนี้จะทำให้ผู้เขียนยูสเคสต้องหลีกเลี่ยงการใช้สัญลักษณ์ หรือเครื่องหมายที่ทำให้ผู้ใช้งานเกิดความสับสนได้ จำนวนยูสเคสที่เกิดขึ้นอาจจะมีจำนวนมากหรือน้อยนั้นขึ้นอยู่กับวิธีการพัฒนาตามความต้องการของระบบ ดังนั้นบางระบบสามารถพัฒนาได้ด้วยการใช้ยูสเคสจำนวนน้อยยูสเคสก็สามารถสร้างระบบงานได้อย่างมีประสิทธิภาพ

ตัวอย่างการเขียนยูสเคสของระบบต่าง ๆ

- 1) ระบบสั่งซื้อสินค้าออนไลน์
- 2) ระบบขายปลีก
- 3) ระบบจองห้องพัก
- 4) ระบบคลินิกทันตกรรม
- 5) ระบบห้องสมุด

ตัวอย่างระบบที่ 5.1 ระบบการสั่งซื้อสินค้าออนไลน์ (Purchase Order Online)



การวิเคราะห์ระบบสั่งซื้อสินค้าออนไลน์

ระบบสั่งซื้อสินค้าออนไลน์ เป็นการสั่งซื้อสินค้าผ่านระบบอินเทอร์เน็ต โดยมีการชำระเงินผ่านบัตรเครดิตเป็นหลัก โดยปกติแล้วผู้ใช้ระบบต้องผ่านการลงทะเบียนก่อนจึงสามารถสั่งซื้อสินค้าได้ รูปแบบการลงทะเบียนจะเป็นการกรอกแบบฟอร์มข้อมูลที่เกี่ยวข้องกับผู้ใช้ ได้แก่ ชื่อ นามสกุล เพศ อีเมล วันเดือนปีเกิด อาชีพ ที่อยู่ เบอร์โทรศัพท์ติดต่อ ชื่อผู้ใช้งานและ รหัสผ่าน เป็นต้น เมื่อผู้ใช้กรอกข้อมูลครบถ้วนและถูกต้องแล้ว โดยชื่อผู้ใช้งาน (User name) จะ ต้องไม่ซ้ำกับในระบบเพื่อใช้ล็อกอินเข้าระบบต่อไป ระบบสั่งซื้อสินค้าออนไลน์มีการจัดเก็บข้อมูลแยกตามหมวดหมู่ของสินค้า ส่วนการนำเสนอรายละเอียดของสินค้าประกอบด้วย รหัสสินค้า ชื่อสินค้า ราคาต่อหน่วย เป็นต้น นอกจากนั้นแล้วผู้ใช้งานสามารถค้นหาสินค้าที่ต้องการได้โดยการระบุคำสำคัญ เพื่อเพิ่มความสะดวกในการเลือกซื้อสินค้าที่ต้องการได้อย่างรวดเร็ว ผู้ใช้สามารถเลือกสั่งซื้อสินค้าได้จากรายการสินค้าในหมวด (Categories) ที่เลือกไว้โดยการคลิกที่ตะกร้าสินค้า (Shopping cart) ระบบจะเลือกสินค้ารายการดังกล่าวเข้าสู่ตะกร้าสั่งซื้อสินค้าที่มีการแสดงรายละเอียดของสินค้า พร้อมทั้งคำนวณค่าใช้จ่าย

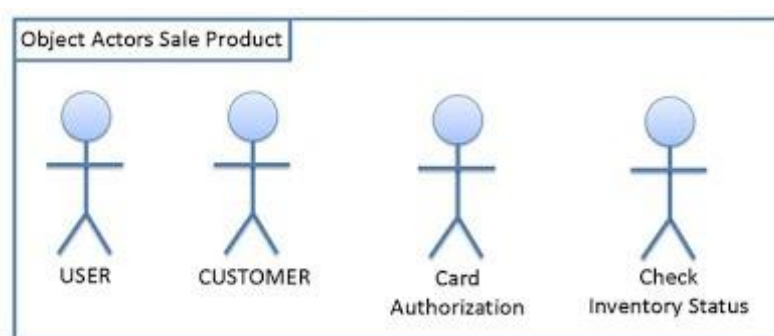
ต่างๆ เช่น ราคารวมแต่ละรายการ ราคารวมทั้งสิ้น ภาษี และราคารวมภาษี เป็นต้น ในขั้นตอนนี้ผู้ใช้สามารถแก้ไขจำนวนสินค้าภายในตะกร้าสินค้าที่ได้มีการสั่งซื้อไปแล้ว หรือ ลบสินค้าที่ไม่ต้องการออกจากตะกร้าสินค้าได้ โดยระบบจะคำนวณค่าใช้จ่ายต่าง ๆ ทุกครั้งเมื่อมีการเปลี่ยนแปลงเกิดขึ้นภายในตะกร้าสินค้า

เมื่อผู้ใช้เสร็จสิ้นการเลือกซื้อสินค้าแล้วขั้นตอนต่อไปจะเป็นการชำระเงินด้วยบัตรเครดิต ซึ่งต้องผ่านการล็อกอินเสมอผู้ใช้ต้องกรอกแบบฟอร์มที่เกี่ยวข้องกับข้อมูลบัตรเครดิต ได้แก่ชนิดของบัตรเครดิต หมายเลขบัตรชื่อเจ้าของบัตร และวันที่หมดอายุเพื่อให้ระบบส่งข้อมูลการชำระเงินไปยังหน่วยงานตรวจสอบสถานะบัตรเครดิตภายนอกเมื่อมีการตรวจสอบสถานะบัตรเครดิตถูกต้องขั้นตอนของการสั่งซื้อสินค้าถือว่าเสร็จสิ้นนอกจากนั้นแล้วเพื่อความสะดวกในการใช้งานของระบบผู้ใช้สามารถตรวจสอบสถานะของการสั่งซื้อสินค้าได้ โดยผ่านการล็อกอินระบบจะแสดงข้อมูลการสั่งซื้อสินค้าทั้งหมดที่ผู้ใช้เคยสั่งซื้อไว้โดยผู้ใช้สามารถตรวจสอบรายการสั่งซื้อสินค้าที่ต้องการได้โดยระบบจะแสดงรายการสินค้าพร้อมทั้งรายละเอียดต่างๆรวมไปถึงสถานะของการจัดส่ง

การเขียนแผนภาพยูสเคสสำหรับระบบสั่งซื้อสินค้าออนไลน์ มีขั้นตอนดังนี้

1. กำหนดแอกเตอร์ (Actor)

เมื่อพิจารณาจากความต้องการแล้วพบว่าผู้ใช้ระบบมี 2 ประเภทคือ ผู้ใช้ทั่วไปที่ต้องการเข้าชมสินค้าภายในระบบ และลูกค้าที่ต้องการสั่งซื้อสินค้าภายในระบบ นอกจากนั้นแล้วในระบบนี้ยังเกี่ยวข้องโดยตรงกับหน่วยงานตรวจสอบสถานะของบัตรเครดิต (Card Authorization) เพื่อ ทำหน้าที่ตรวจสอบสถานะทางการเงินของลูกค้า ส่วนการตรวจสอบจำนวนสินค้าในคลังเพื่อใช้ประกอบสั่งซื้อสินค้านั้น เป็นหน้าที่ของระบบตรวจสอบสต็อกสินค้า (Check Inventory System) ดังนั้นแอกเตอร์ของระบบได้แก่ ผู้ใช้ ลูกค้า ระบบตรวจสอบสถานะของบัตรเครดิต และ ระบบตรวจสอบสต็อกสินค้า



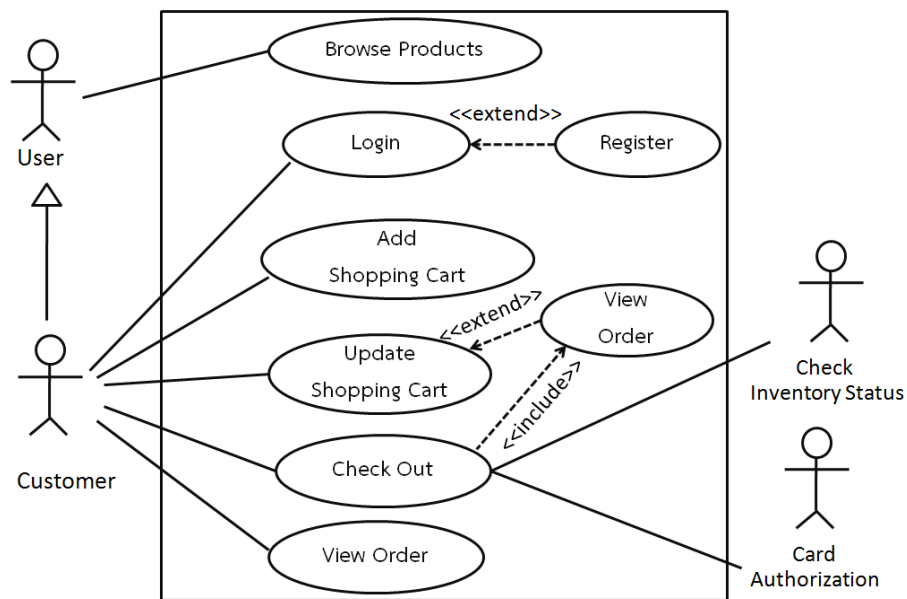
ภาพที่ 5.16 แสดงแอกเตอร์ของระบบสั่งซื้อสินค้าออนไลน์

2. กำหนดยูสเคส (Use Case)

การกำหนดยูสเคสของระบบสั่งซื้อสินค้าออนไลน์ ผู้ใช้สามารถค้นหารายละเอียดของสินค้า (Browse Product) ได้ด้วยตนเอง ในกรณีของการพิจารณาพบว่าแอกเตอร์คือลูกค้า จะต้องมีการลงทะเบียน

(Register) เพื่อบันทึกข้อมูลที่จำเป็นในการลงทะเบียนรวมทั้งชื่อผู้ใช้งานและรหัสผ่านสำหรับเข้าสู่ระบบ (Log in) โดยลูกค้าสามารถเลือกสินค้าที่ต้องการเข้าสู่การสั่งซื้อ (Add Shopping Cart) และเมื่อสั่งซื้อสินค้าแล้วจะสามารถตรวจสอบยอดรวมและราคาสินค้าที่ปรากฏในรถเข็นได้ (View Shopping Cart) นอกจากนี้หากพบข้อผิดพลาดจากการสั่งซื้อ ลูกค้าสามารถแก้ไขสินค้าในรถเข็น (Update Shopping Cart) ที่ได้สั่งซื้อไปแล้ว

นอกจากนั้นระบบจะคำนวณค่าใช้จ่ายต่าง ๆ เช่น ราคารวมในแต่ละรายการ ราคารวมทั้งสิ้น และภาษี ซึ่งลูกค้าจะต้องกรอกรายละเอียดการชำระเงินและกรอกสถานที่จัดส่งสินค้า เพื่อทำการส่งสินค้าให้แก่ลูกค้าต่อไป (Check Out) ในกรณีที่ยังไม่ได้รับสินค้าตามระยะเวลาที่กำหนดลูกค้าจะต้องสามารถตรวจสอบสถานะของสินค้า (View Order Status) ที่ต้องจัดส่งให้แก่ลูกค้าได้



ภาพที่ 5.17 แผนภาพยูสเคสของระบบสั่งซื้อสินค้าออนไลน์

3. การเขียนคำอธิบายยูสเคส (Use Case Description) ของระบบสั่งซื้อสินค้าออนไลน์

Actor : ลูกค้า (Customer)

Use Case : เข้าสู่ระบบ (Log in)

Actor Action	System Response
1. เมื่อลูกค้าเลือกฟังก์ชันล็อกอิน	
2. ลูกค้าระบุชื่อผู้ใช้งานและรหัสผ่าน	
	3. ระบบทำการตรวจสอบความครบถ้วนของข้อมูล
	4. ระบบรับค่าชื่อผู้ใช้งานและรหัสผ่าน
	5. ระบบค้นหาชื่อผู้ใช้งานและรหัสผ่านในฐานข้อมูล

	6. ระบบคืนค่าสถานะของการตรวจสอบจากฐานข้อมูล
	7. ระบบตรวจสอบสถานะของการคืนค่า
	8. ระบบยอมให้ลูกค้าเข้าสู่ระบบ
	8.1 การชำระเงิน
	8.2 ดูข้อมูลที่เคยส่งไปทั้งหมด
9. ลูกค้าจะเห็นหน้าจอสำหรับการใช้งาน	

Alternative flow of log in events	
ขั้นตอนที่ 3 - ในกรณีที่ลูกค้ากรอกข้อมูลไม่ครบระบบจะแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ” และทำขั้นตอนที่ 2 ซ้ำ	
ขั้นตอนที่ 7 - ในกรณีที่การตรวจสอบชื่อผู้ใช้งานและรหัสผ่านไม่ถูกต้อง ระบบจะแสดงข้อความ “ข้อมูลไม่ถูกต้อง กรุณาตรวจสอบอีกครั้ง” และกลับสู่หน้าจอล็อกอิน	

Actor: ลูกค้า (Customer) และ พนักงาน (Check inventory status)

Use Case: เลือกสินค้าลงในรถเข็นสินค้า (Add cart shopping)

Actor	System Response
1. ผู้ใช้ค้นหารายละเอียดของสินค้า	
	2. ระบบพิจารณาพบว่าแอดเดอคือลูกค้า
	3. ระบบเข้าสู่หน้าลงทะเบียนเพื่อบันทึกข้อมูลที่จำเป็นในการลงทะเบียน ชื่อผู้ใช้งานและรหัสผ่านสำหรับเข้าสู่ระบบ
4. เข้าสู่ระบบสมาชิก	
5. ลูกค้าเลือกสินค้าที่ต้องการเข้าสู่การสั่งซื้อ	
	6. สั่งซื้อสินค้าแล้วระบบจะโชว์การตรวจสอบยอดรวมและราคาสินค้าที่ปรากฏในรถเข็นได้
7. หากพบข้อผิดพลาดจากการสั่งซื้อ ลูกค้าสามารถแก้ไขสินค้าในรถเข็น	
	8. ระบบจะคำนวณค่าใช้จ่ายต่างๆ
	9. ระบบจะโชว์รายละเอียดของสินค้าที่สั่งได้

10. ยืนยันการสั่งซื้อ	
	11. ระบบจะตรวจสอบสถานะบัตรเครดิตของลูกค้า
12. เจ้าหน้าที่เช็คสต็อกสินค้าและทำการจัดส่ง	

Alternative flow of add cart shopping events
กรณีที่ถูกค้ายังไม่ได้รับสินค้าตามระยะเวลาที่กำหนด
ขั้นตอนที่ 9 ลูกค้าจะต้องสามารถตรวจสอบสถานะของสินค้า (View Order Status)

ตัวอย่างการเขียน

การสร้างแผนภาพยูสเคสเพื่ออธิบายการจองห้องพักซึ่งเกิดจากผลของการวิเคราะห์ความต้องการเบื้องต้น สามารถเขียนเป็นรายการได้ดังนี้

ความต้องการ

ในแต่ละวันจะมีการจองห้องพักของนักท่องเที่ยว โดยนักท่องเที่ยวที่จองห้องพักในแต่ละวันจะมี 2 ประเภทคือ

- 1) นักท่องเที่ยวใหม่
- 2) นักท่องเที่ยวปัจจุบันที่ต้องการอยู่ต่อ
- 3) การจองห้องพักในแต่ละครั้งจะมีการเก็บหลักฐานและค่ามัดจำห้องพัก
- 4) ซึ่งการจองห้องพักจะเสร็จสิ้นได้ก็ต่อเมื่อมีการยืนยันออนไลน์หรือทำการเช็คอินที่หน้าห้องล๊อบบี้
- 5) ในขณะที่เดียวกันเงินค่าห้องพักที่เรียกเก็บได้ก็ต้องมีจำนวนครบถ้วนด้วย
- 6) พนักงานของโรงแรมจะเป็นผู้จัดการในเรื่องของการจัดเก็บหลักฐานและค่าห้องพักทั้งหมด
- 7) ผู้จ่ายเงินต้องเป็นนักท่องเที่ยวเท่านั้น
- 8) สำหรับนักท่องเที่ยวบางคนที่ได้รับสิทธิพิเศษเช่น
 - 8.1) ได้รับบัตรกำนัลห้องพักฟรี
 - 8.2) ได้รับบัตรกำนัลส่วนลดค่าห้องพัก
 - 8.3) จะมีสิทธิได้รับยกเว้นค่าห้องพักในบางวัน

แอกเตอร์ของระบบคือ

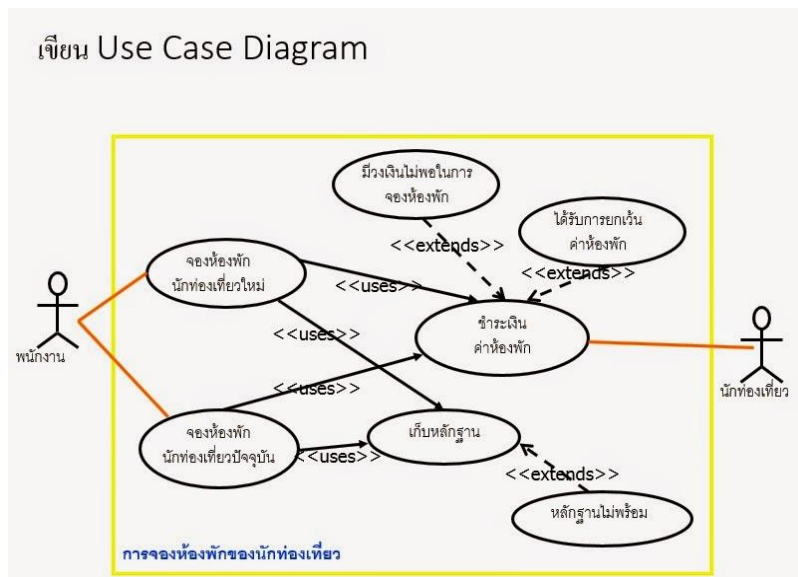
- 1) พนักงาน
- 2) นักท่องเที่ยว

ยูสเคสของระบบคือ

- 1) การจองห้องพักนักท่องเที่ยว
- 2) การเก็บหลักฐานต่าง ๆ
- 3) การชำระค่าห้องพัก

ยูสเคสที่เกี่ยวข้อง

- 1) การจองห้องพักของนักท่องเที่ยว
 การจองห้องพักของนักท่องเที่ยวใหม่
 การจองห้องพักของนักท่องเที่ยวปัจจุบันที่ต้องการอยู่ต่อ
- 2) การเก็บหลักฐาน
 หลักฐานต่าง ๆ ไม่พร้อม
- 3) การชำระค่าห้องพัก
 มีวงเงินไม่พอในการจองห้องพัก
 ได้รับการยกเว้นค่าห้องพัก



ภาพที่ 5.18 ยูสเคสสำหรับระบบการจองห้องพัก

ตัวอย่างระบบที่ 5.6 ระบบการจัดการคลินิกทันตกรรม



ระบบการจัดการคลินิกทันตกรรม (Dental clinic management system)

1. วิเคราะห์ระบบการจัดการคลินิกทันตกรรม

คลินิกทันตกรรมทั่วไปทันตกรรมจัดฟันโดยในแต่ละวันจะมีคนไข้มาทำการรักษาอย่างต่อเนื่องและเพิ่มมากขึ้นเรื่อยๆทำให้พื้นที่การเก็บเอกสารประวัติของคนไข้เพิ่มมากขึ้นเช่นเดียวกันปัจจุบันการจัดการคลินิกทันตกรรมนั้นจะมีการจัดการเอกสารต่าง ๆ เป็นแบบทำด้วยมือจะเป็นกระบวนการทำบัตรผู้ป่วย การเก็บบันทึกประวัติผู้ป่วยการนัดหมายผู้ป่วย การเก็บข้อมูลยา และข้อมูลอุปกรณ์ทางการแพทย์การเก็บข้อมูลการรักษาและการส่งจ่ายยาจะเก็บลงในแฟ้มเอกสารจริงซึ่งยังไม่มีเทคโนโลยีใด ๆ มาใช้เพื่ออำนวยความสะดวกซึ่งทำให้เกิดความล่าช้าในการค้นหาประวัติของผู้ป่วยและต้องใช้พื้นที่ในการจัดเก็บเอกสารเป็นจำนวนมากทำให้เอกสารชำรุดเกิดการสูญหายของข้อมูลได้ค่าชำระรักษาพยาบาลปัจจุบันยังไม่มีการออกใบเสร็จให้กับผู้ป่วยเวลาผู้ป่วยต้องการที่จะเรียกดูข้อมูลการรักษาของผู้ป่วยเองทางคลินิกจึงต้องค้นหาเอกสารของผู้ป่วยรายนั้นเป็นเวลานานทำให้การทำงานเป็นไปด้วยความล่าช้าขาดประสิทธิภาพและเกิดความผิดพลาดได้จากปัญหาที่พบดังกล่าวจึงมีแนวคิดที่นำเอาเทคโนโลยีสารสนเทศเข้ามาใช้ในการบริหารจัดการคลินิกเพื่อให้ระบบมีความสะดวกรวดเร็ว มีประสิทธิภาพต่อผู้ป่วยที่มาทำการรักษาในคลินิกทั้งยังให้ความสะดวกสบายต่อผู้ดูแลระบบและยังช่วยลดเวลาในการทำงานให้น้อยลง

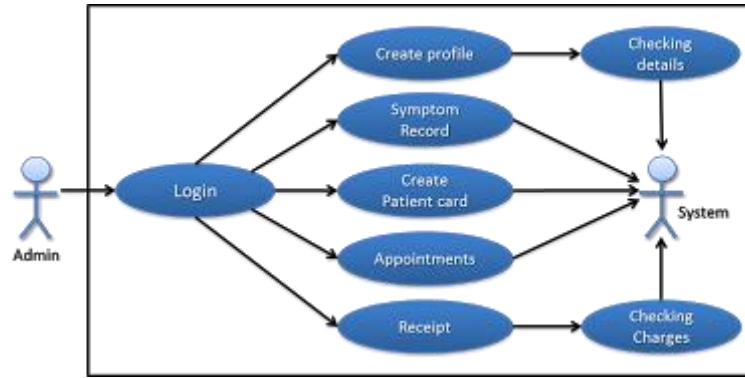
2. การสร้างแผนภาพยูสเคสสำหรับระบบการจัดการคลินิกทันตกรรม

1. กำหนดแอกเตอร์

ผู้ดำเนินการในระบบประกอบด้วย (Admin) หรือ พนักงาน

2. กำหนดยูสเคส

การกำหนดยูสเคสของระบบการจัดการคลินิกพนักงาน สามารถเข้าสู่ระบบถ้าเป็นคนไข้ที่ยังไม่เคยได้รับการบริการให้สร้างแบบฟอร์มข้อมูลผู้ป่วย (Create Profile) และมีการตรวจสอบความถูกต้องของรายละเอียด (Checking Details) ก่อนเข้าระบบฐานข้อมูล เมื่อผู้ป่วยได้รับการตรวจ พนักงานสามารถบันทึกอาการของผู้ป่วย (Symptom Record) เพื่อเก็บประวัติการรักษาเข้าสู่ระบบ พนักงานออกบัตรผู้ป่วย (Create Patient Card) ให้กับผู้ป่วย เมื่อมีการตรวจรักษาอย่างต่อเนื่อง พนักงานจึงต้องออกใบหมอนัด (Appointments) ในครั้งถัดไป หลังจากเสร็จสิ้นการเข้ารับการรักษาแล้วพนักงานจะต้องออกใบเสร็จชำระเงิน (Receipt) และต้องตรวจสอบค่าใช้จ่ายทั้งหมด (Checking Charges) (Tutorialspoint, 2014)



ภาพที่ 5.19 แผนภาพยูสเคสของระบบการจัดการคลินิกทันตกรรม

3. การเขียนคำอธิบายยูสเคสระบบการจัดการคลินิกทันตกรรม

แอกเตอร์ คือ เจ้าหน้าที่พนักงาน

ยูสเคสการเข้าสู่ระบบ

แอกเตอร์	System Response
1. เข้าสู่หน้าเพจของระบบจัดการคลินิกทันตกรรม	
	2. แสดงส่วนล็อกอินเข้าสู่ระบบสำหรับ Admin
3. Admin กรอก username และ password	
	4. ตรวจสอบสิทธิ์การเข้าใช้ระบบ
5. เมื่อตรวจสอบสิทธิ์ผ่านจึงจึงทำรายการอื่นๆได้	

Actor : เจ้าหน้าที่พนักงาน (Admin)

Use case : บันทึกอาการผู้ป่วย (Symptom record)

Actor	System response
1. เข้าสู่เพจของระบบจัดการคลินิกทันตกรรม	
	2. แสดงแบบฟอร์มบันทึกอาการผู้ป่วย
3. กรอกข้อมูลตามแบบฟอร์ม	
	4. ตรวจสอบความถูกต้อง
5. ยืนยันบันทึกอาการผู้ป่วย	
	6. บันทึกข้อมูลลงฐานข้อมูล

Actor: เจ้าหน้าที่พนักงาน (Admin)

Use case: ออกบัตรประจำตัวผู้ป่วย (Create partial card)

Actor	System Response
1. เข้าสู่เพจของระบบการจัดการคลินิกทันตกรรม	
	2. แสดงแบบฟอร์มประวัติของผู้ป่วย
3. กรอกข้อมูลตามแบบฟอร์ม	
	4. ตรวจสอบความถูกต้อง
5. ยืนยันการออกบัตรประจำตัวผู้ป่วย	
	6. ออกบัตรประจำตัวผู้ป่วย

Actor : เจ้าหน้าที่พนักงาน (Admin)

Use case : ออกใบนัดหมายผู้ป่วย (Appointments)

Actor	System Response
1. เข้าสู่เพจของระบบการจัดการคลินิกทันตกรรม	
	2. แสดงแบบฟอร์มการนัดหมายผู้ป่วย
3. นัดหมายผู้ป่วยจากปฏิทินในแต่ละเดือน	
	4. ตรวจสอบวันนัดหมาย
5. ยืนยันการนัดหมายผู้ป่วย	
	6. บันทึกนัดหมายผู้ป่วย

Actor : เจ้าหน้าที่พนักงาน (Admin)

Use case : ออกใบเสร็จค่าใช้จ่ายของผู้ป่วย (Receipt)

Actor	System Response
1. เข้าสู่เพจของระบบการจัดการคลินิกทันตกรรม	
	2. แสดงแบบฟอร์มค่าใช้จ่ายของผู้ป่วย
3. บันทึกค่าใช้จ่ายของผู้ป่วย	
	4. ตรวจสอบค่าใช้จ่ายผู้ป่วย
5. ยืนยันค่าใช้จ่ายผู้ป่วย	
	6. ออกใบเสร็จชำระเงิน

ตัวอย่างที่ 5.6 ระบบการยืมหนังสือในห้องสมุด (Borrow book from library)



1. วิเคราะห์ระบบการยืมหนังสือในห้องสมุด

ผู้ที่ทำการยืมหนังสือให้ทำการกรอกรหัสสมาชิกเข้ามาเพื่อตรวจสอบการเป็นสมาชิก ถ้าเป็นก็ตรวจสอบอีกว่าสมาชิกมีข้อมูลการจองหนังสือหรือไม่ถ้ามีก็แสดงข้อมูลการจองหนังสือและทำการยืมได้ถ้าหากไม่มีก็ตรวจสอบอีกว่าต้องการยืมหนังสือเล่มอื่นหรือไม่ถ้าต้องการก็ทำการกรอกรหัสประจำหนังสือและทำการยืมหนังสือได้แล้วตรวจสอบว่าต้องการยืมหนังสืออีกหรือไม่ ถ้าต้องการก็ให้กลับไปกรอกข้อมูลหนังสือ เพื่อทำการยืมต่ออีก แต่ถ้าไม่ก็บันทึกข้อมูลการยืมและพิมพ์ใบสลีปการยืมจบการทำงาน

2. การวาดแผนภาพยูสเคสสำหรับระบบการยืมหนังสือในห้องสมุด

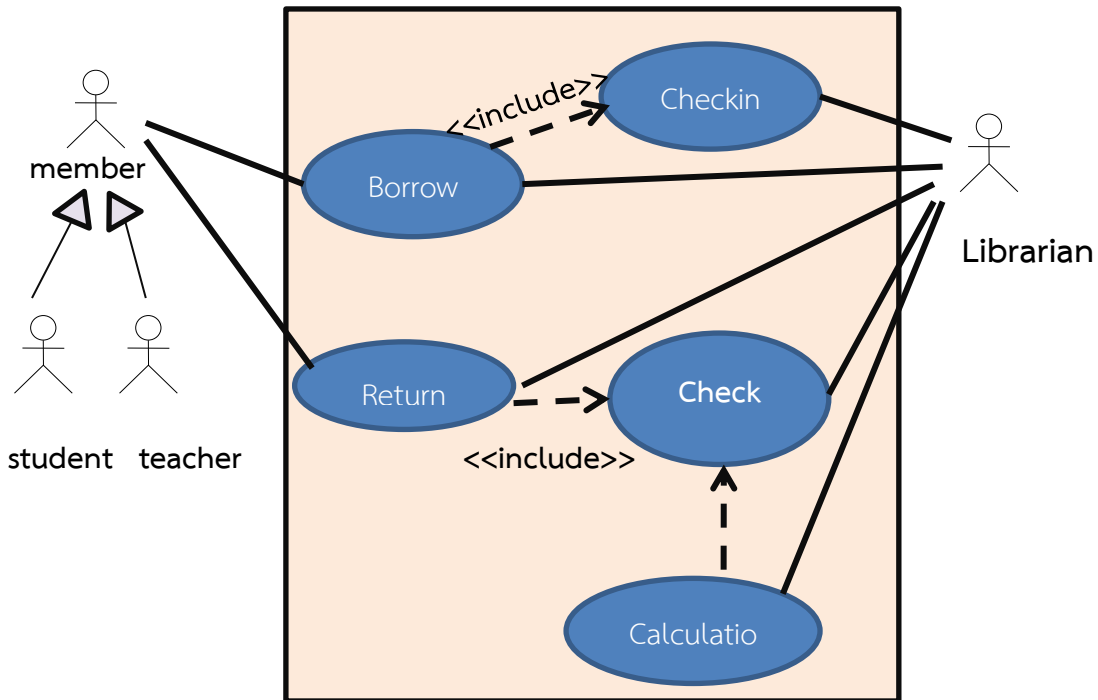
1. กำหนดแอกเตอร์

ผู้ดำเนินการในระบบประกอบด้วย สมาชิก (Member) และ บรรณารักษ์ (Librarian)

2. กำหนดยูสเคส

ยูสเคสในระบบการยืมหนังสือห้องสมุด นักเรียน และอาจารย์ สามารถเป็นสมาชิกได้ทำการยืมหนังสือ (Borrow book) ต้องผ่านการตรวจสอบสมาชิก (Checking member) จากบรรณารักษ์ (Librarian) ก่อนว่ามีการยืมค้างอยู่หรือไม่ถ้าไม่มีก็สามารถยืมหนังสือได้ ส่วนการคืนหนังสือ (Return book) ต้องผ่านการตรวจสอบระยะเวลาในการยืม (Checking period borrowed) ก่อนว่ามีการยืมเกินเวลาหรือไม่ ถ้าเกินเวลา ระบบจะคิดค่าปรับ (Calculation fine) ที่ยืมเกินระยะเวลาที่กำหนด

ระบบการยืมหนังสือในห้องสมุด (Borrow book from library)



ภาพที่ 5.20 แผนภาพยูสเคสของระบบการยืมหนังสือในห้องสมุด

3. การเขียน Use Case Description ระบบการยืมหนังสือ

Actor : สมาชิก (Member) บรรณารักษ์ (Librarian)

Use case : การยืมหนังสือ (Borrow book)

Member	Librarian
1.สมาชิกทำการยืมหนังสือ	
	2. บรรณารักษ์ทำการตรวจสอบสมาชิกว่ามี สถานะการยืมค้างอยู่หรือไม่
	2.1 ถ้ามีการค้างสถานะการยืมให้ทำการคืนก่อน
	2.2 ถ้าไม่มีการค้างสถานะการยืม
3. สามารถยืมหนังสือได้	

Actor: สมาชิก (Member) บรรณารักษ์ (Librarian)

Use case: การยืมหนังสือ (Borrow book)

Member	Librarian
1. สมาชิกทำการคืนหนังสือ	
	2. บรรณารักษ์ทำการตรวจสอบสมาชิกว่าสมาชิกยืมเกินเวลาหรือไม่

	2.1 ถ้าเกินเวลาต้องทำการเสียค่าปรับ
3. สามารถคืนหนังสือได้	

สรุป

การวิเคราะห์ความต้องการเป็นสิ่งสำคัญลำดับต้น ๆ ของการพัฒนาระบบโดยใช้หลักการเชิงวัตถุ เป็นการหาความต้องการของผู้ใช้ระบบโดยใช้เทคนิคการหาความต้องการแบบต่าง ๆ สิ่งที่คุณพัฒนาต้องทำในขั้นตอนนี้ คือ การนำเอาความต้องการแบบ Functionality (Functional Requirements) ซึ่งบ่งบอกถึงความสามารถที่ผู้ใช้สามารถเรียกใช้จากระบบ มาแปลงเป็นโมเดลความต้องการของผู้ใช้งาน ซึ่งจะใช้แผนภาพยูสเคส (Use case diagram) ที่ขาดไม่ได้ในยูสเคส คือคำบรรยายยูสเคส (Use Case Description) ซึ่งเป็นรายละเอียดของแต่ละฟังก์ชันว่าเริ่มต้นอย่างไร มีการดำเนินเหตุการณ์เกิดขึ้นอย่างไรและสิ้นสุดลงอย่างไร เรียกรวม ๆ ว่า ลำดับการเกิดเหตุการณ์ (Flow of Event) รวมถึงเหตุการณ์ยกเว้น (Exception Flow of Event) ที่อาจเกิดขึ้นระหว่างการปฏิบัติฟังก์ชันดังกล่าวของระบบ ก็ต้องถูกบันทึกด้วยเช่นกัน เพื่อใช้เป็นข้อมูลพื้นฐานก่อนนำไปวิเคราะห์ และออกแบบระบบในขั้นตอนต่อไป

ชื่อ-นามสกุล	รหัส	สาขาวิชา	รุ่น/หมู่	คะแนน	ลายเซ็นต์ อาจารย์

แบบฝึกหัดท้ายบทที่ 5

1. จงตอบคำถามต่อไปนี้

1.1 ยูสเคส “การถอนเงินที่เคาน์เตอร์ธนาคาร” กับ ยูสเคส “การตรวจสอบจำนวนเงินในบัญชีของผู้ถอนเงิน” ควรมีความสัมพันธ์แบบใด

.....

.....

1.2 ยูสเคส “การสร้างบ้าน” กับ ยูสเคส “การปรับเปลี่ยนแปลนของบ้าน” ควรมีความสัมพันธ์แบบใด

.....

.....

1.3 ยูสเคส “การซื้อตั๋วรถไฟโดยซื้อจากเครื่องขายตั๋ว” กับ ยูสเคส “การซื้อตั๋วรถไฟจากเจ้าหน้าที่” ควรมีความสัมพันธ์แบบใด

.....

.....

2. ให้สร้าง System Requirement Model ของระบบจากรายละเอียดข้างล่างนี้

2.1 การซื้อขายเงินตราต่างประเทศของสถาบันการเงิน จำแนกเป็น การซื้อขายเช็คเดินทางและการแลกเปลี่ยนเงินต่างสกุล ซึ่งการแลกเปลี่ยนเงินตราต่างประเทศนี้จำแนกออกเป็น การเปลี่ยน ณ เวลานั้น และการแลกเปลี่ยนล่วงหน้า โดยลูกค้าที่สามารถแลกเปลี่ยนเงินตราต่างประเทศแบบแลกเปลี่ยนล่วงหน้าได้นั้น ต้องเป็นลูกค้าที่มีบัญชีกับธนาคาร

2.2 ในการซื้อขายเงินตราแต่ละครั้งระบบต้องสามารถทำการคำนวณมูลค่าเงินที่จะซื้อจะขายแต่ละสกุลได้ โดยพิจารณาจากอัตราแลกเปลี่ยนของวันนั้นๆ

2.3 รายการซื้อขายเงินตราต่างประเทศอาจไม่สามารถดำเนินการได้ หากเงินบางสกุลที่ลูกค้าต้องการซื้อหรือแลกเปลี่ยน ไม่มีอยู่ในคลังของธนาคาร

2.4 แต่สำหรับการซื้อขายเช็คเดินทางจะไม่ประสบปัญหา

2.5 ทุก ๆ ประเภทรายการของการซื้อขายเงินตราต่างประเทศ ลูกค้าต้องจ่ายเงินค่าธรรมเนียมในการทำธุรกรรมให้แก่ธนาคารเสมอ โดยรายการแต่ละประเภทจะมีอัตราค่าธรรมเนียมและวิธีการคิดค่าธรรมเนียมที่แตกต่างกันออกไป

3. จงสร้างแบบจำลองยูสเคสจากรายละเอียดของระบบการเข้าพักห้องพักเริ่มจากผู้ใช้ค้นหาข้อมูลห้องว่าง ตรวจสอบสถานะของห้องว่างหรือไม่ จากเพิ่มข้อมูลห้องพัก ถ้ามีห้องว่างก็จะทำสัญญา ถ้าไม่มีห้องว่างก็ทำการจอง รับชำระเงินค่าประกันทำการบันทึกและออกใบเสร็จ จบการทำงาน เจ้าของห้องพักจะได้รับรายงาน ข้อมูลห้องพักและข้อมูลการเข้าพัก

4. จงสร้างแบบจำลองยูสเคสจากรายละเอียดของระบบโรงงานแห่งหนึ่งประกอบด้วยระบบการผลิต ระบบควบคุมคุณภาพ ซึ่งทำหน้าที่ในการควบคุมระบบการผลิต โดยในระบบการผลิตจะมีคนงานเป็นผู้ปฏิบัติงาน นอกจากนี้ โรงงานยังมีระบบการตลาดทำหน้าที่ขายผลิตภัณฑ์ต่างๆ ของโรงงาน โดยระบบการตลาดและระบบควบคุมคุณภาพจะมีผู้จัดการทำหน้าที่จัดการ

5. จงสร้างแบบจำลองยูสเคสจากรายละเอียดของระบบการซื้อขายเงินตราต่างประเทศของสถาบันการเงิน จำแนกเป็น การซื้อขายเช็คเดินทางและการแลกเปลี่ยนเงินต่างสกุล การแลกเปลี่ยนเงินต่างสกุล จำแนกออกเป็น การแลกเปลี่ยน ณ เวลานั้น และการแลกเปลี่ยนล่วงหน้า

6. จงสร้างแบบจำลองยูสเคสของระบบห้องสมุดในมหาวิทยาลัยแห่งหนึ่ง ซึ่งผู้เข้าใช้บริการได้แก่ นักศึกษา อาจารย์ และพนักงานของมหาวิทยาลัย ในกระบวนการทำงาน เจ้าหน้าที่ห้องสมุดจะทำหน้าที่ให้บริการยืมหรือคืนหนังสือของแก่บุคคลดังกล่าว นอกจากนี้ยังจะต้องจัดการกับทรัพยากรในห้องสมุด เช่น การเพิ่มแก้ไข ลบข้อมูลหนังสือและวารสาร รวมทั้งข้อมูลเกี่ยวกับสมาชิกของห้องสมุดอีกด้วย ทุกเดือน เจ้าหน้าที่ห้องสมุดจะต้องทำรายงานต่าง ๆ ส่งให้กับผู้อำนวยการศูนย์สารสนเทศ เช่น รายงานจำนวนสมาชิกรายงานจำนวนหนังสือและวารสารใหม่รายงานการยืมหรือคืนหนังสือ รายงานค่าปรับล่าช้า เป็นต้น

เอกสารอ้างอิง

- กิตติพงษ์ กลมกล่อม. (2552). *การวิเคราะห์และออกแบบระบบเชิงวัตถุด้วย UML*. กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- กิติ ภัคตีวัฒนกุล, และกิตติพงษ์ กลมกล่อม. (2544). *UML: วิเคราะห์และออกแบบระบบเชิงวัตถุ*. กรุงเทพฯ: เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- دنۇۋىس ائسزنانن تۇگۇل. (2555). *การวิเคราะห์และออกแบบเชิงวัตถุ*. สืบค้น 11 กรกฎาคม 2563, จาก <https://sites.google.com/site/danuwat154/ooad>
- นัฐพงศ์ ส่งเนียม. (2563). *วิชาวิเคราะห์และออกแบบระบบเชิงวัตถุ (OOAD)*. สืบค้น 18 กันยายน 2563, จาก http://www.siam2dev.net/siam2dev_4122506_OOAD.php
- พนิดา พานิชกุล. (2548). *Object-Oriented ฉบับพื้นฐาน*. กรุงเทพฯ: เคทีพีคอมพ์แอนดคอนซัลท์จำกัด.
- Object Management Group. (1997). *WHAT IS UML*. Retrieved 18 September 2020, from <https://www.uml.org/>
- Shin, S. (2015). A Study on the Difficulties of Learning Phase Transition in Object-Oriented Analysis and Design From the Viewpoint of Semantic Distance. *IEEE Transactions on Education*, 58(2), 117-123.
- Tutorialspoint. (2014). *Object Oriented Analysis & Design Tutorial*. Retrieved 18 September 2020, from https://www.tutorialspoint.com/object_oriented_analysis_design/index.htm
- Ritabluerayit solutions. (2014). Joint Application Development (JAD) Definition. Retrieved 21 January 2019, from <https://ritabluerayitsolutions.wordpress.com/2014/03/01/154/>