

System Modeling

อ. ประจักษ์ ปิยะวงศ์วิศาล

Pratch Piyawongwisal

Today's Agenda

- Preparing for final presentation *สำคัญ*
- Github + Trello Updates
- Requirement Engineering Recap & Quiz
- Scrum Wrap-up
 - Review/Retrospective meeting
 - Burndown chart
- System Modeling
 - Diagrams

สำคัญ Final Project Presentation

- สัปดาห์สุดท้ายก่อนสอบ (12 มี.ค.) ทุกกลุ่มจะต้องนำเสนอโครงงานหน้าชั้นเรียน
- กิจกรรมนี้เป็นการจำลอง **review meeting** ในวิธี **Scrum**
 - ส่งงานที่เสร็จบางส่วนให้แก่ **stakeholders** (สมมติว่าลูกค้าในที่นี้คืออาจารย์และเพื่อนกลุ่มอื่น)
 - นำ **feedback** ไปปรับปรุงใน **sprint** ถัดไป
- คิดเป็นคะแนน **10% => สำคัญมาก**

* สำคัญ * Final Project Presentation

แนวทางปฏิบัติ

- ใช้เวลานำเสนอกลุ่มละประมาณ **15** นาที + Q&A อีก **5** นาที
- ทุกคนในกลุ่มจะต้องได้พูด
- มีการ **demo Ionic App** โดยละเอียด
 - ก่อนนำเสนอจะต้องทดลองรันแอปให้มั่นใจว่าทำงานได้ถูกต้อง รวมถึงตรวจสอบในส่วน **firebase**
 - ให้ **deploy app** ขึ้น **Heroku** โดยทำตาม <https://devdactic.com/deploy-ionic-website-heroku/> เพื่อให้ **stakeholder** จะได้ทดลองใช้จริงได้
- หลังนำเสนอจบ กลุ่มอื่นจะต้องถามคำถาม หรือมีข้อเสนอแนะอย่างน้อย **1** อย่าง

* สำคัญ * Final Project Presentation

เนื้อหาใน **slides** (ขั้นต่ำ)

- ที่มาของโปรเจค ปัญหาที่ต้องการแก้ไข กลุ่มผู้ใช้เป้าหมาย
- การแบ่งหน้าที่และความรับผิดชอบ (**task** ใน **scrumboard**)
- รายการ **requirement, user stories** ที่ทำเสร็จ/ไม่เสร็จ
- **tools** ที่ใช้พัฒนา
- อุปสรรค สิ่งที่จะปรับปรุงใน **sprint** ถัดไป
- **link** ไปหน้า **demo, github, trello**
- อื่นๆ

GitHub + Trello Updates

- **GitHub** จะต้องมี
 - เอกสารที่จัดทำทั้งหมดใน **folder /doc**
 - ภาพ **Screenshot** ของ **Trello** รายสัปดาห์
 - **Source code** ของ **Ionic**
 - การ **commit** จากสมาชิกทุกคน !
- **Trello**
 - มีการอัปเดตสม่ำเสมอ (ทุก **1-2** วัน)
 - ตรงกับความจริงใน **GitHub**
 - ทุกคนมี **task** ที่ตนรับผิดชอบ เท่าเทียมกัน
 - คอลัมน์ **Done** จะต้องแนบหลักฐาน **screenshot** ในส่วนที่ทำเสร็จ

Requirement Engineering Recap

Requirement Engineering

- กระบวนการระบุข้อกำหนด (**requirement**) คือคำอธิบายเกี่ยวกับ **service** ต่าง ๆ ที่ผู้ใช้ต้องการจากระบบ รวมถึงขีดจำกัด (**constraint**) ต่าง ๆ ของระบบ
- ประเภทของ **requirement** โดยแบ่งตาม **target audience**
 - User requirements สำหรับลูกค้า, คร่าว ๆ เน้นเข้าใจง่าย, ใช้ภาษาพูด + แผนภาพ
 - System requirements สำหรับ **dev**, ลงรายละเอียด, ใช้ภาษาทางเทคนิค
- ประเภทของ **requirement** โดยแบ่งตามสิ่งที่ระบุ
 - Functional requirements ฟังก์ชันการทำงานของระบบ เช่น **user** ต้องสามารถค้นหาหนังสือได้
 - Non-Functional requirements ขีดจำกัด (**constraint**) ต่าง ๆ เช่น **failure rate**

Requirement Engineering Process

- Requirement elicitation
 - ทำการ **interview** สอบถามความต้องการจาก **stakeholder** โดยใช้ **user stories, use case** เป็นเครื่องมือ
- Requirement analysis and specification
 - วิเคราะห์แล้วลงมือเขียนทั้ง **User requirement** และ **System requirement**
- Requirement validation
 - ตรวจสอบว่า **requirement** ตรงความต้องการลูกค้าจริง, ไม่มี **conflict** ไต ๆ ระหว่างแต่ละ **requirement**, มีความเป็นไปได้, สามารถทดสอบได้
- Requirement management
 - ติดตาม **requirement** ที่ทำสำเร็จ, การเปลี่ยนแปลงของ **requirement**

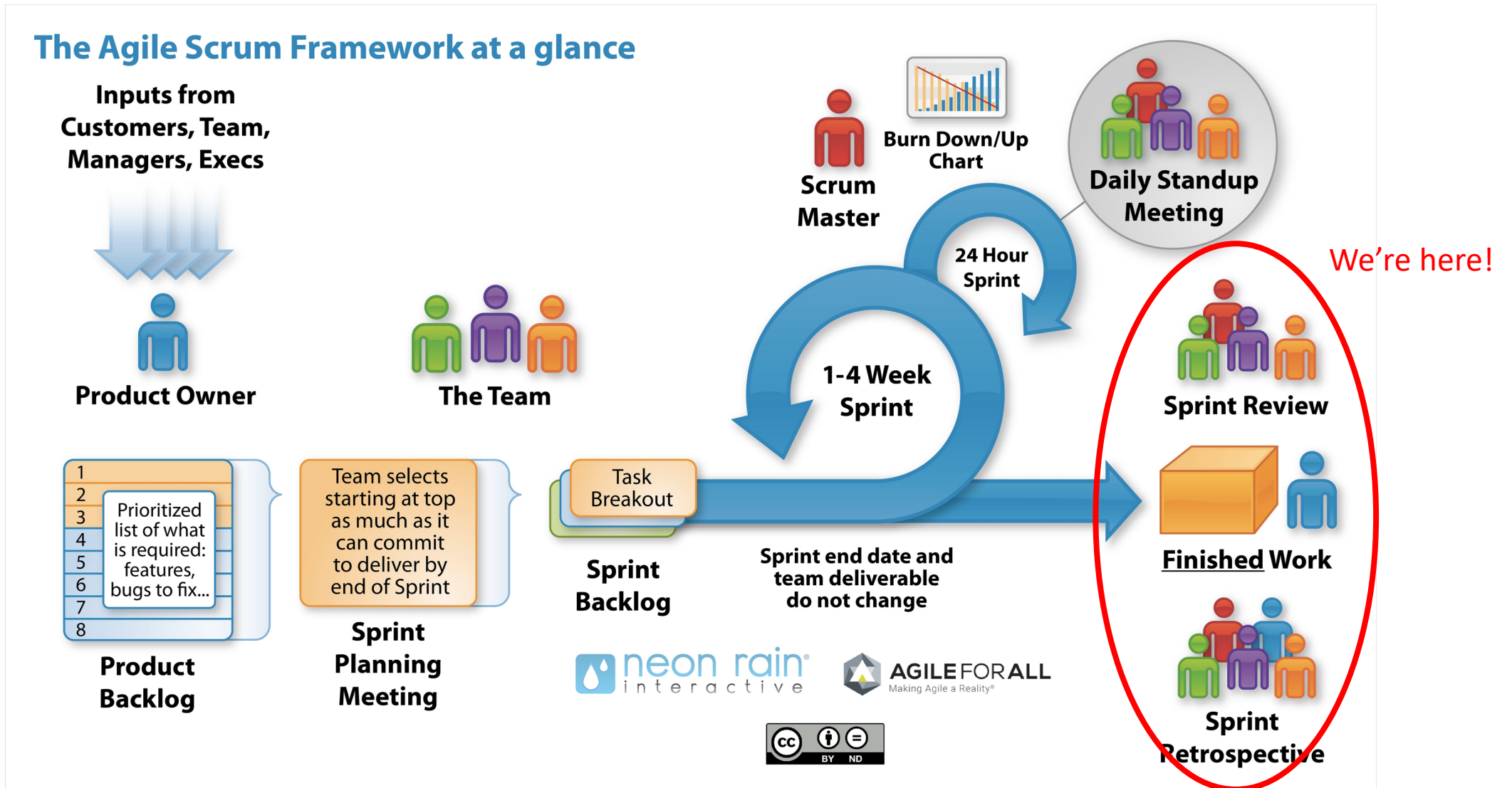
ลองคิดว่าใน **scrum** เราได้ทำกระบวนการเหล่านี้อย่างไร โดยใช้ **artifact** ไตบ้าง

Requirement Engineering Quiz

1. กระบวนการ Requirement Engineering มีความสำคัญต่อผู้ที่มีบทบาทใดในวิธีการ Scrum มากที่สุด
2. การเขียน Requirement ในรูป "As a... I want... so that..." ถือว่าเป็น Requirement แบบใด User Requirement หรือ System Requirement
3. ข้อจำกัดด้าน response time ของเซิร์ฟเวอร์ ถือว่าเป็น requirement แบบ functional หรือ non-functional
4. การเก็บ requirement ด้วยวิธีสัมภาษณ์ ถือว่าอยู่ในขั้นตอนใด (elicitation/specification/validation/management)
5. ในวิธีการ scrum การนำ feedback ของลูกค้ามาปรับแต่ง Product Backlog ในกิจกรรม Retrospective Meeting ถือว่าเป็นขั้นตอนใด (elicitation/specification/validation/management)

Scrum Wrap-up

Scrum



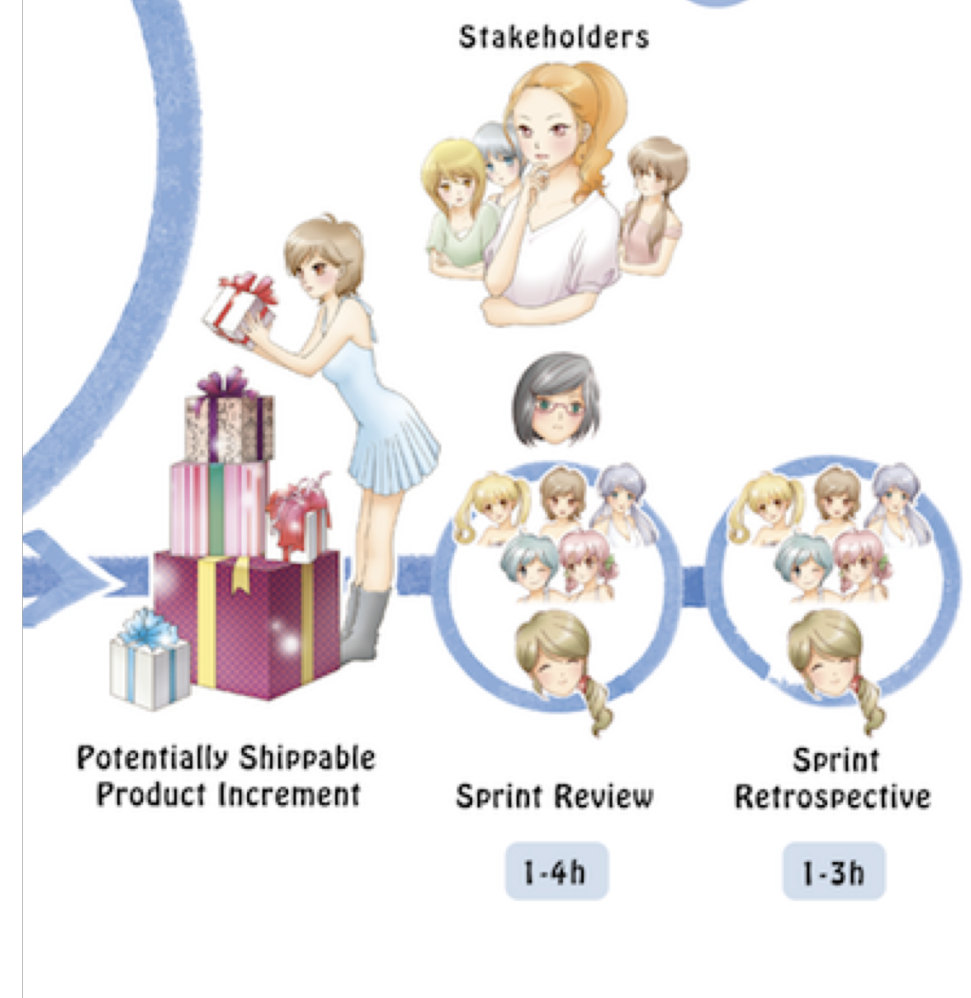
End of Sprint

1. Review meeting

- **Players:** ทุกคน + stakeholders
- demo งานที่เสร็จ + พูดถึงงานที่เหลือ
- นำ feedback จาก stakeholder ไปปรับ product backlog สำหรับ Sprint ถัดไป

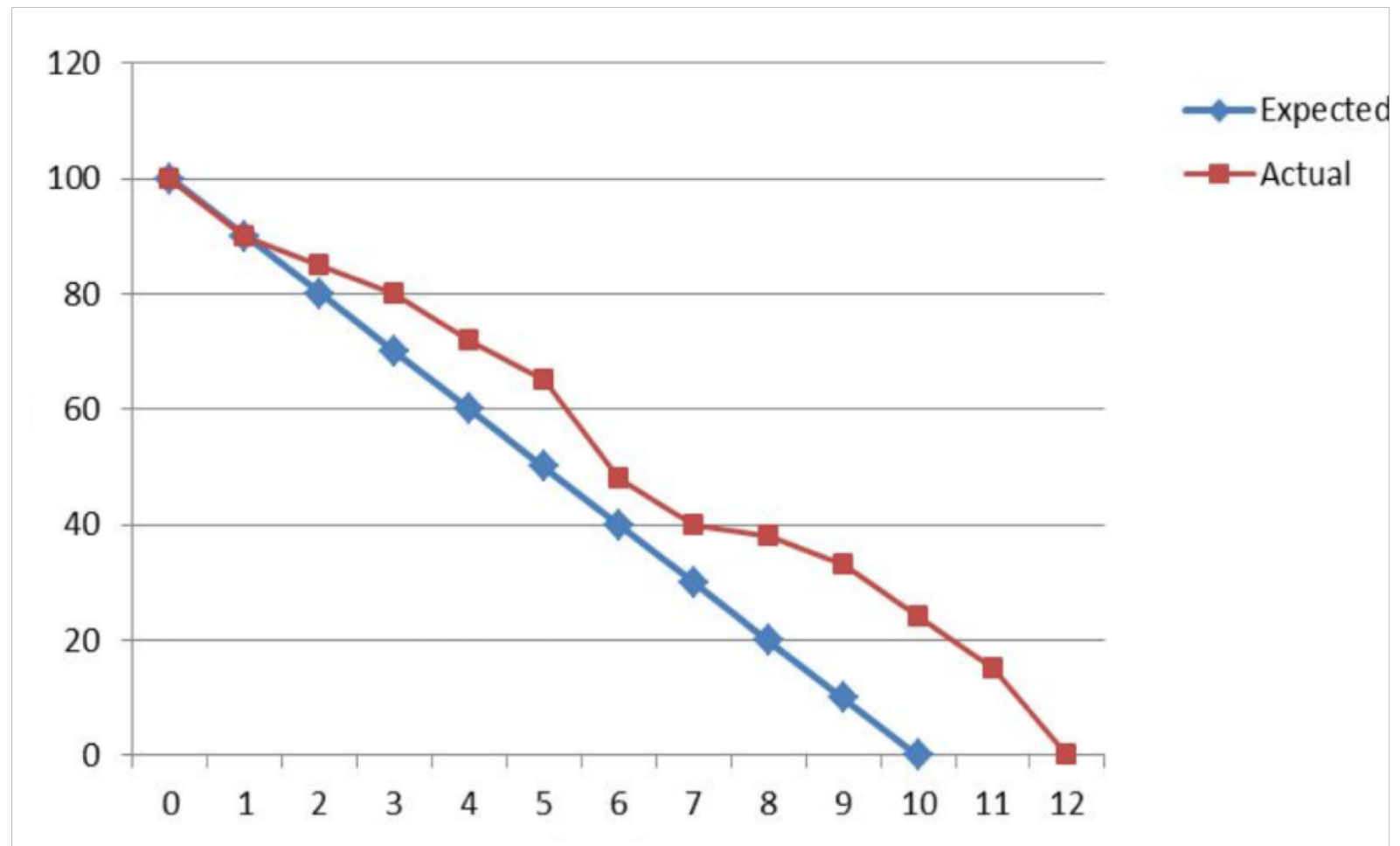
2. Retrospective meeting

- **Players:** scrum master, dev
- ต่อจาก review meeting ทันที
- หาวิธีถึงการปรับปรุง process



Burndown Chart

- plots remaining work vs days



Example Scrum Worksheet

ชื่อ Senior Project (ภาษาไทย)

ชื่อ Senior Project (ภาษาอังกฤษ)

ขั้นตอนที่ 1 Product Backlog

เขียน Requirements ของระบบอย่างน้อย 4 ความต้องการ ลงในตาราง

ID	Requirements	User Stories
1		As a I can So that
2		As a I can So that
3		As a I can So that
4		As a I can So that

ขั้นตอนที่ 2 Sprint Planning

ให้กำหนด Priority และ Story Point ลงในตาราง (ID คือหมายเลขเดียวกับขั้นตอนที่ 1)

ID	Priority	Story Point

ขั้นตอนที่ 3 Sprint Backlog

เขียนการจัดลำดับของ Sprint ที่ได้จากขั้นตอน Sprint Planning ลงในตาราง

(Sprint ที่ n คือ รอบของการทำ Sprint และ ID คือหมายเลขเดียวกับขั้นตอนที่ 1)

Sprint ที่ n	ID	Task To Do	วันส่งมอบงาน

ขั้นตอนที่ 4 Daily Scrum

ให้อธิบายการทำงานของขั้นตอนนี้ พร้อมทั้งยกตัวอย่างสถานการณ์จำลองของ Project ที่ทำ

ขั้นตอนที่ 5 Sprint Review and Retrospective

ให้อธิบายการทำงานของขั้นตอนนี้ พร้อมทั้งยกตัวอย่างสถานการณ์จำลองของ Project ที่ทำ

ขั้นตอนที่ 6 Estimation (10 คะแนน)

เขียน Burndown Chart เพื่อประเมินการทำงานทั้งกระบวนการในการพัฒนาซอฟต์แวร์ที่วางแผนไว้ตั้งแต่วันเริ่มต้นทำงานถึงวันที่ส่งมอบงานครบ ซึ่งกราฟที่ได้จะต้องมีความสอดคล้องกับข้อมูลในขั้นตอนที่ 2 และ 3 (กำหนดให้ แกน x คือ เวลา และ แกน y คือ จำนวน Story Points ของ Release ที่ยังทำไม่เสร็จ ณ เวลาใดๆ)

ชื่อ Senior Project (ภาษาอังกฤษ) A Support Tool for Teaching

ขั้นตอนที่ 1 Product Backlog

เขียน Requirements ของระบบอย่างน้อย 4 ความต้องการ ลงในตาราง

ID	Requirements	User Stories
1	ซอฟต์แวร์มีการ Check ชื่อนักศึกษาที่เข้าห้องเรียน นักศึกษาที่ส่งการบ้าน ผ่าน Mobile Device	As a ผู้ใช้ I can ตรวจสอบชื่อ การส่งการบ้านได้ So that ให้ทราบสถานะของนักศึกษา
2	มีการระบบการ Chat หรือการส่ง Messages หรือ ไฟล์งาน ระหว่างกัน ผ่าน Mobile Device	As a ผู้ใช้ I can ส่งข้อความสื่อสารระหว่างกัน So that ส่งงานหรือตอบคำถามระหว่างเรียนได้
3	ในแต่ละบทเรียนสามารถทำ Quiz ผ่าน และประมวลผลคะแนนผ่าน Mobile Device	As a ผู้ใช้ I can ทำ Quiz และดูคะแนน So that สะดวกในการทำและตรวจสอบคะแนน
4	สร้างเกมหรือกิจกรรม Workshop ให้ผู้เรียนมีส่วนร่วมผ่าน Mobile Device	As a ผู้ใช้ I can เล่นเกมหรือทำกิจกรรมระหว่างเรียน So that เพิ่มความสนใจในการเรียนของนักศึกษา

Example Scrum Worksheet

ขั้นตอนที่ 2 Sprint Planning

ให้กำหนด Priority และ Story Point ลงในตาราง (ID คือหมายเลขเดียวกับขั้นตอนที่ 1)

ID	Priority	Story Point
1	1	20
2	3	40
3	2	40
4	4	80
5	5	30

Example Scrum Worksheet

ขั้นตอนที่ 3 Sprint Backlog

เขียนการจัดลำดับของ Sprint ที่ได้จากขั้นตอน Sprint Planning ลงในตาราง

(Sprint ที่ n คือ รอบของการทำ Sprint และ ID คือหมายเลขเดียวกับขั้นตอนที่ 1)

Sprint ที่ n	ID	Task To Do	วันส่งมอบงาน
1	1,5	ทำระบบ login, กำหนด Security Policy, ทำ DB, ทำระบบให้ Security และตรวจสอบความถูกต้อง	3 Weeks
2	2	ทำระบบส่งข้อความ ส่งรูป ส่งเสียง	4 Weeks
3	3	ทำแบบทดสอบในแต่ละการเรียนรู้	7 Weeks
4	4	ทำเกมและกิจกรรมให้เข้ากับบทเรียน	10 Weeks

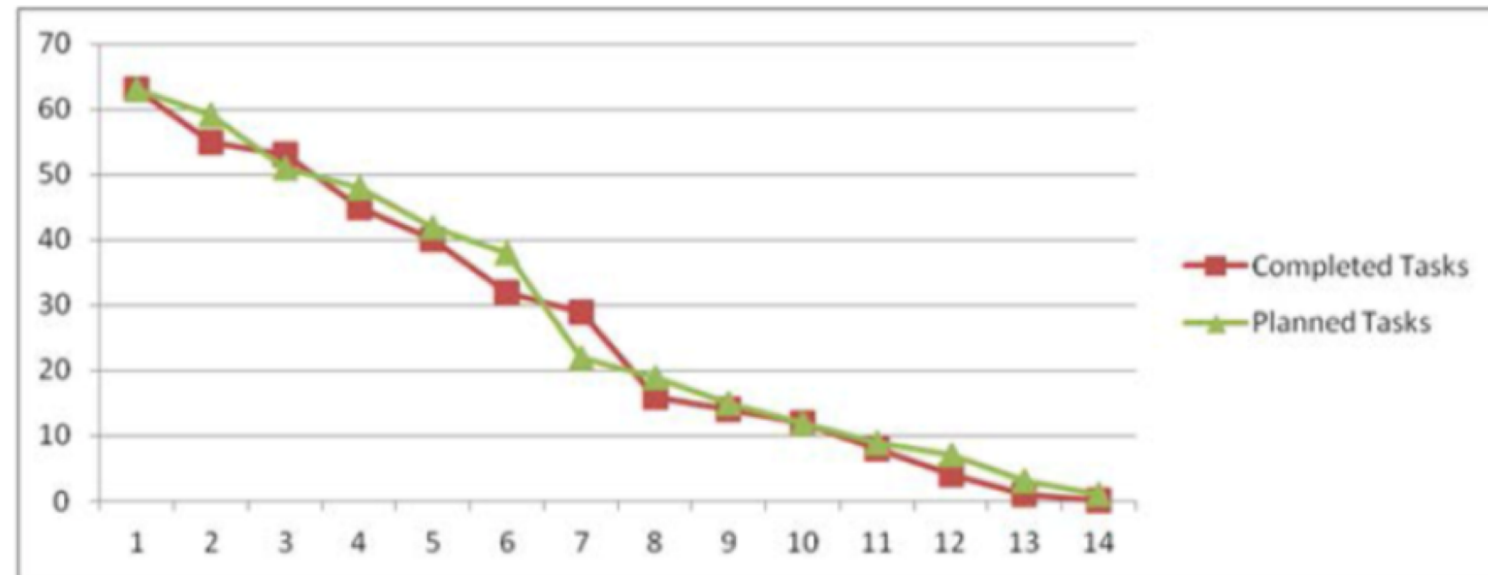
Burndown Chart

ขั้นตอนที่ 6 Estimation

เขียน Burndown Chart เพื่อประเมินการทำงานทั้งกระบวนการในการพัฒนาซอฟต์แวร์ที่วางแผนไว้ตั้งแต่วันเริ่มต้นทำจนถึงวันที่ส่งมอบงานครบ ซึ่งกราฟที่ได้จะต้องมีความสอดคล้องกับข้อมูลในขั้นตอนที่ 2 และ 3

(กำหนดให้ แกน x คือ เวลา และ แกน y คือ จำนวน Story Points ของ Release ที่ยังทำไม่เสร็จ ณ เวลาใดๆ)

ตัวอย่าง



System Modeling

การจำลองระบบ

System Modeling

- การสร้างแบบจำลองเพื่ออธิบายฟังก์ชันการทำงาน (**functionality**) หรือองค์ประกอบ (**component**) ต่าง ๆ ของระบบ
- ทำในช่วง **design phase**
- มักใช้แผนภาพต่าง ๆ เป็นเครื่องมือช่วยในการออกแบบซอฟต์แวร์
- แบบจำลองสามารถอธิบายระบบได้จาก **4** ด้านหลักๆ ดังนี้
 - **External** จำลองบริบท (**context**) หรือสิ่งที่อยู่รอบระบบ
 - **Interaction** จำลองการปฏิสัมพันธ์ระหว่างส่วนต่างๆ ในระบบ
 - **Structural** จำลองโครงสร้างของข้อมูลในระบบ
 - **Behavioral** จำลองพฤติกรรมการทำงานของระบบ

System Modeling

- แผนภาพที่ควรรู้จัก
 - Data Flow Diagram
 - ER Diagram
 - UML Class Diagram
 - UML Use Case Diagram
 - UML Sequence Diagram

กระแสข้อมูลระหว่าง **process** ต่าง ๆ

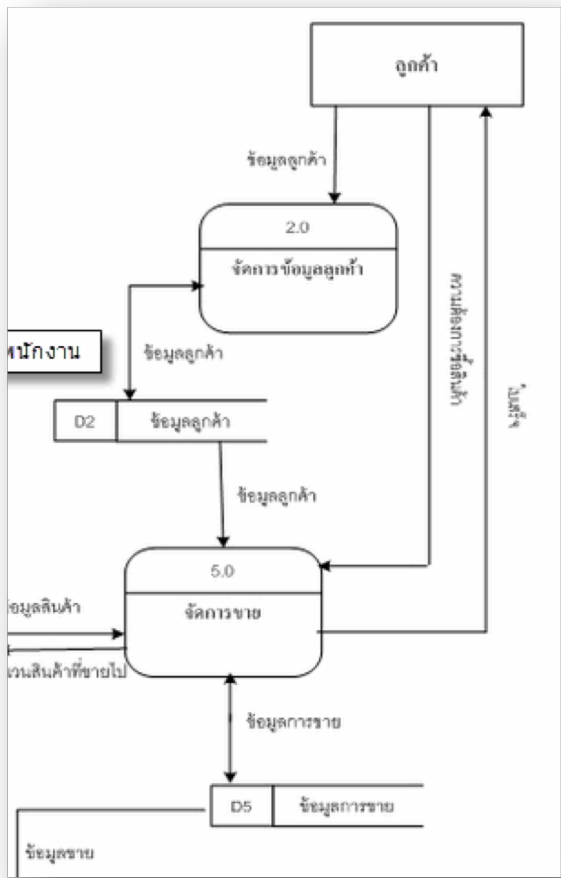
ฐานข้อมูล (ชื่อตาราง, ชื่อคอลัมน์)

คลาสต่าง ๆ ในโปรแกรมเชิงวัตถุ

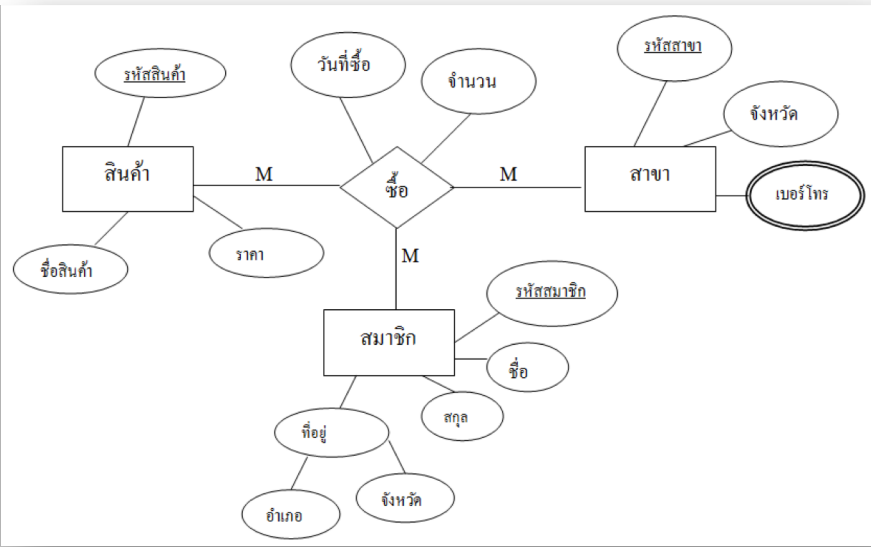
requirement ของลูกค้า

ลำดับการทำงานของระบบ

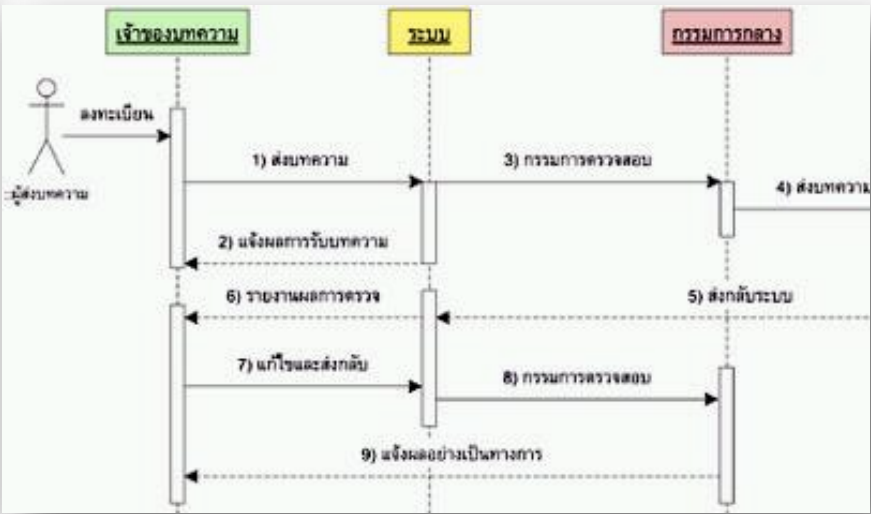
Data Flow Diagram



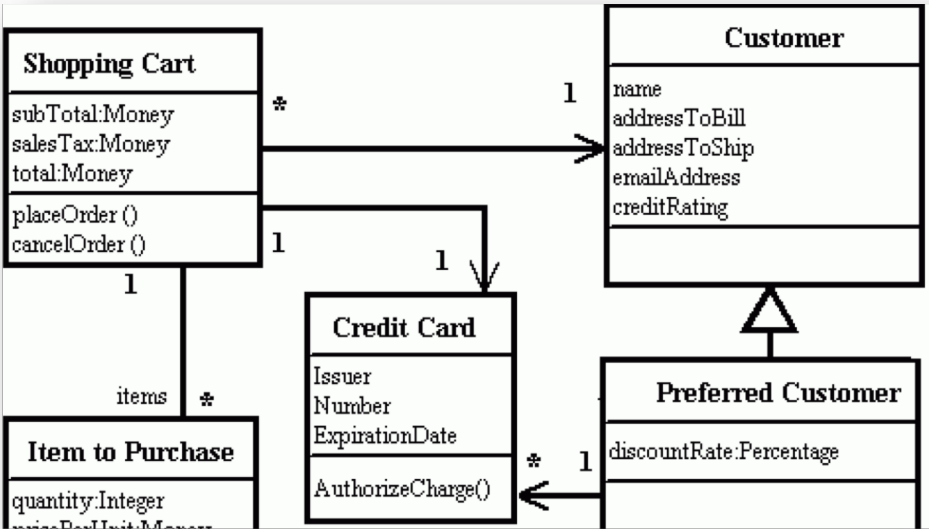
ER Diagram



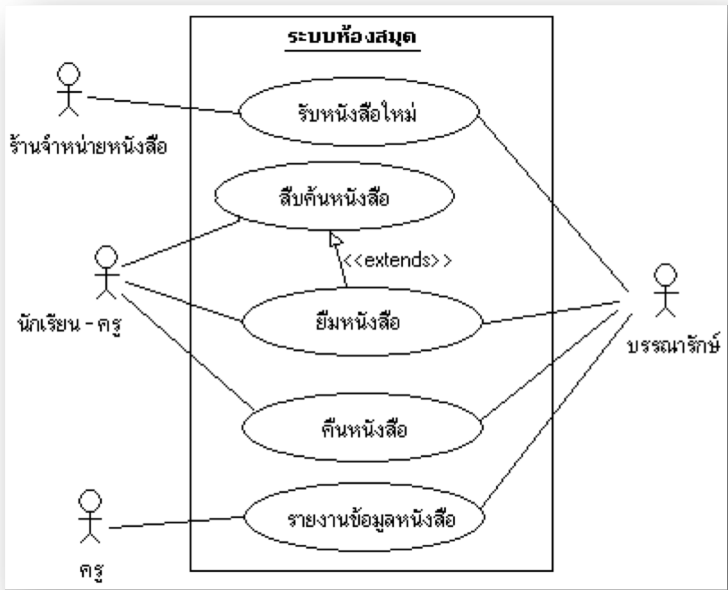
UML Sequence Diagram



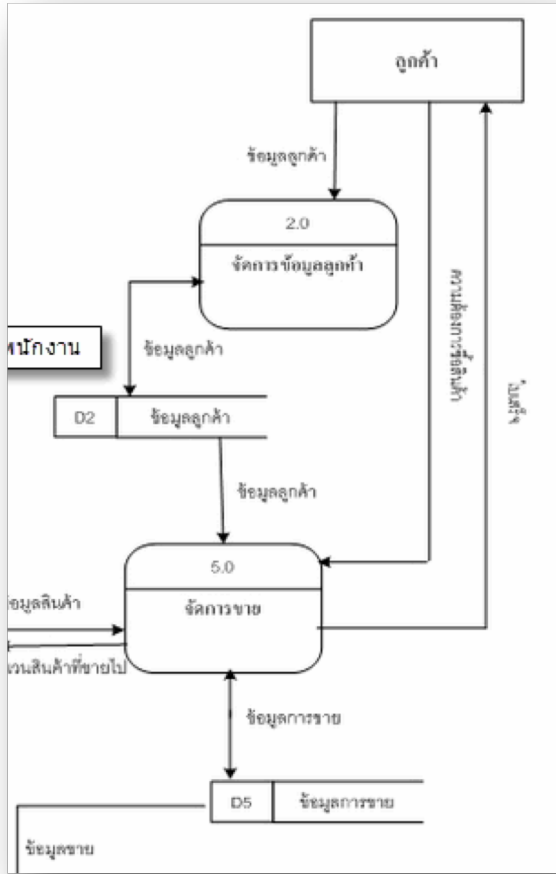
UML Class Diagram



UML Use Case Diagram

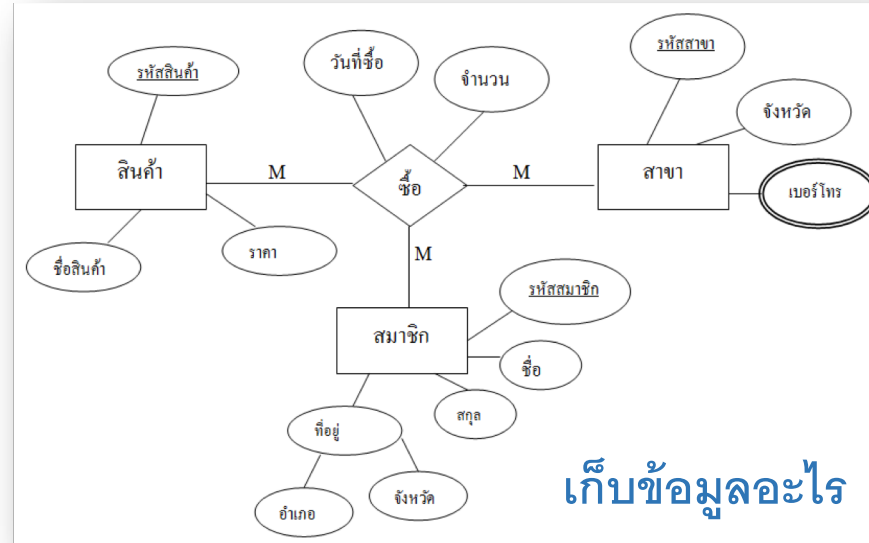


Data Flow Diagram



ข้อมูลไหลไป
process ต่างๆ อย่างไร

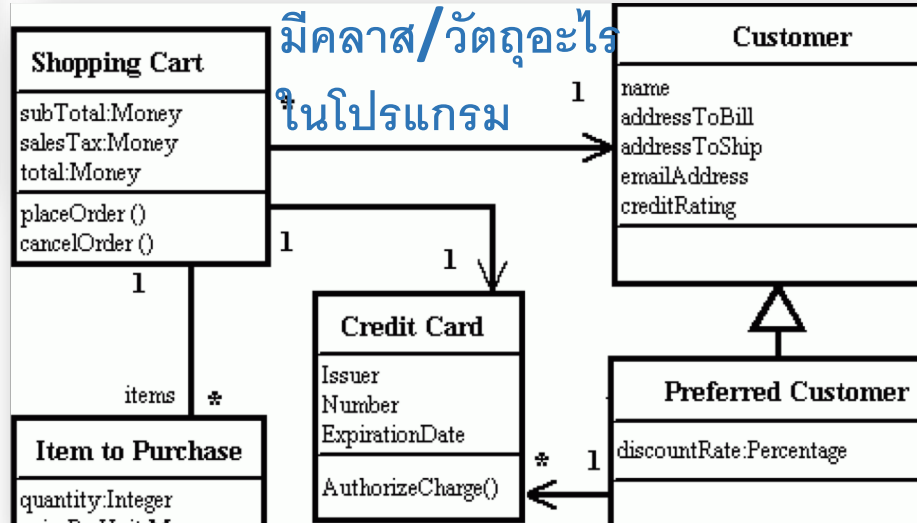
ER Diagram



UML Sequence Diagram



UML Class Diagram



UML Use Case Diagram

