

Current Trend in System Development

01418321 System Analysis and Design
Chalothon Chootong (Ph.D.)

Department of Computer Science and Information, Faculty of
Science at Sriracha, Kasetsart University Sriracha Campus

chootong.c@ku.th



ระบบในปัจจุบัน

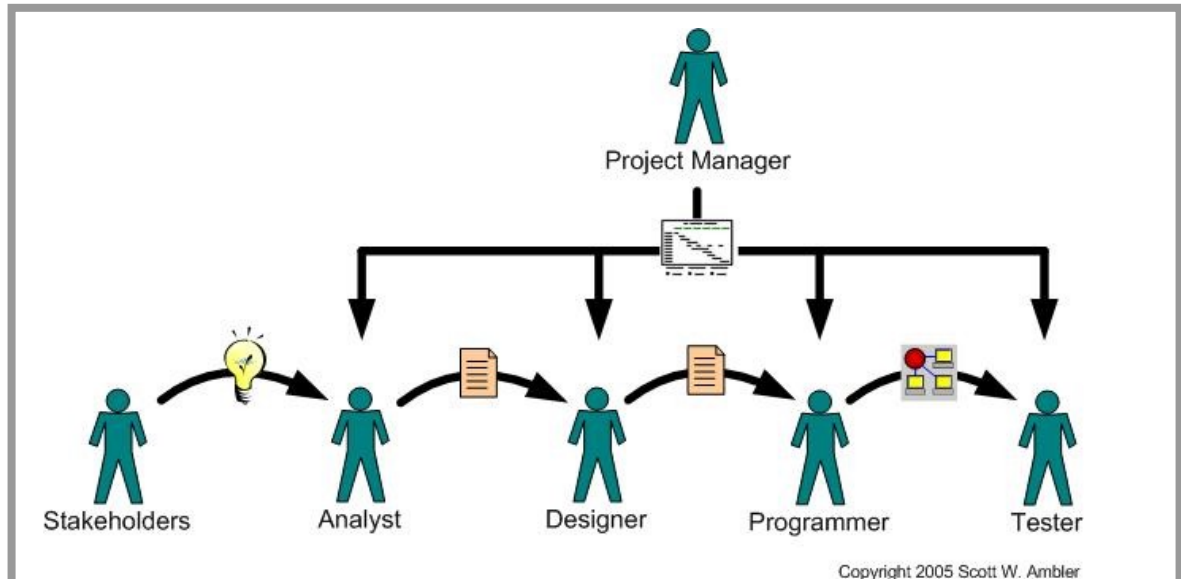
- ▷ ความต้องการของผู้ใช้มักมีการเปลี่ยนแปลงบ่อย
- ▷ เพราะในโลกแห่งธุรกิจย่อมต้องมีการแข่งขัน
- ▷ วิธีการปฏิบัติแบบใหม่ที่สามารถปรับเปลี่ยนให้สอดคล้องกับสภาพแวดล้อม (Adaptive Development)

คุณสมบัติของแนวทางการพัฒนาแบบ Adaptive

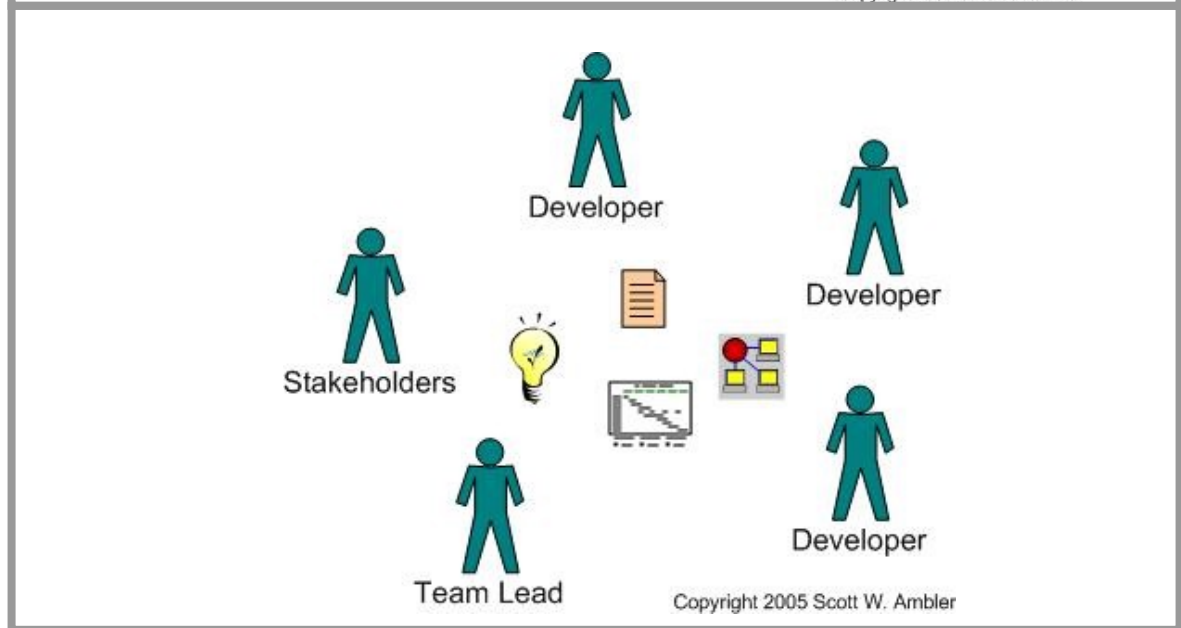
- ▷ ให้ความสำคัญของการวิเคราะห์ ออกแบบ และการจัดทำเอกสาร
น้อยลง
- ▷ มุ่งเน้นที่การพัฒนาระบบ
- ▷ เน้นการมีส่วนร่วมในทีมในการจัดทำโครงการ
- ▷ ลดการวางแผนในรายละเอียด (กรณีการทำงานในขั้นตอนสั้นๆ)
- ▷ จัดการตารางเวลา โดยการแบ่งงานออกเป็นส่วนย่อยๆ ให้เหมาะสมกับ
ขอบเขตเวลา
- ▷ ทำงานร่วมกันภายในทีมเล็กๆ โดยการบริหารจัดการร่วมกัน

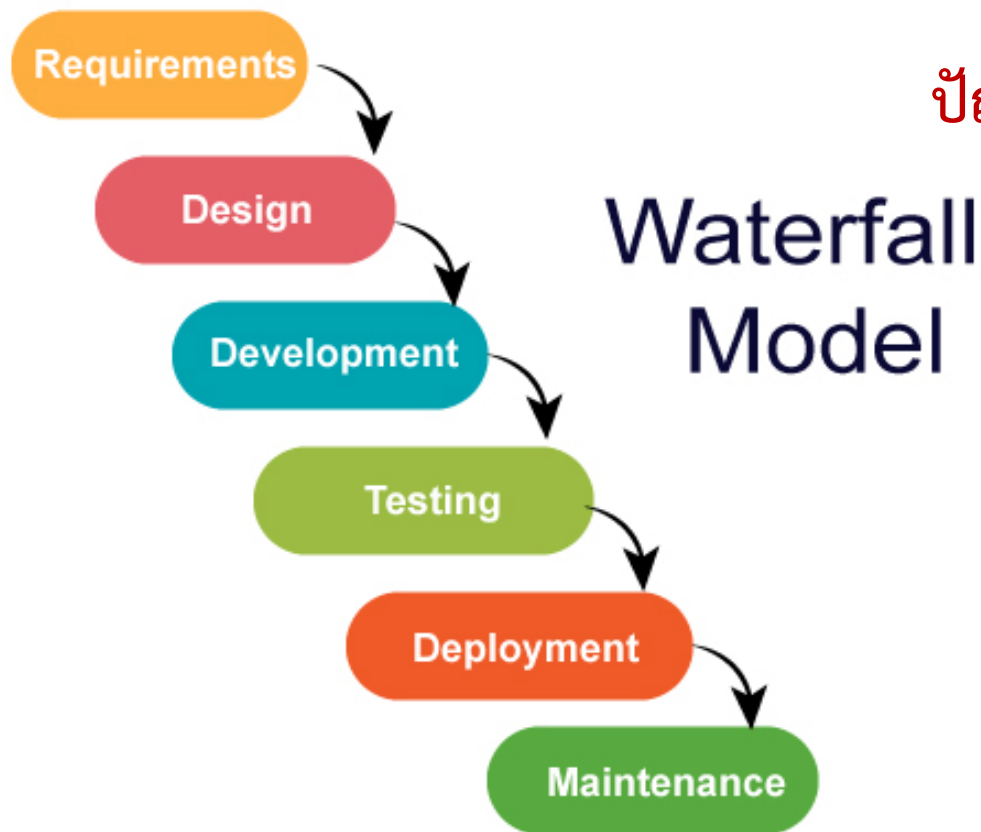
Traditional-team & whole-team collaboration

Traditional team

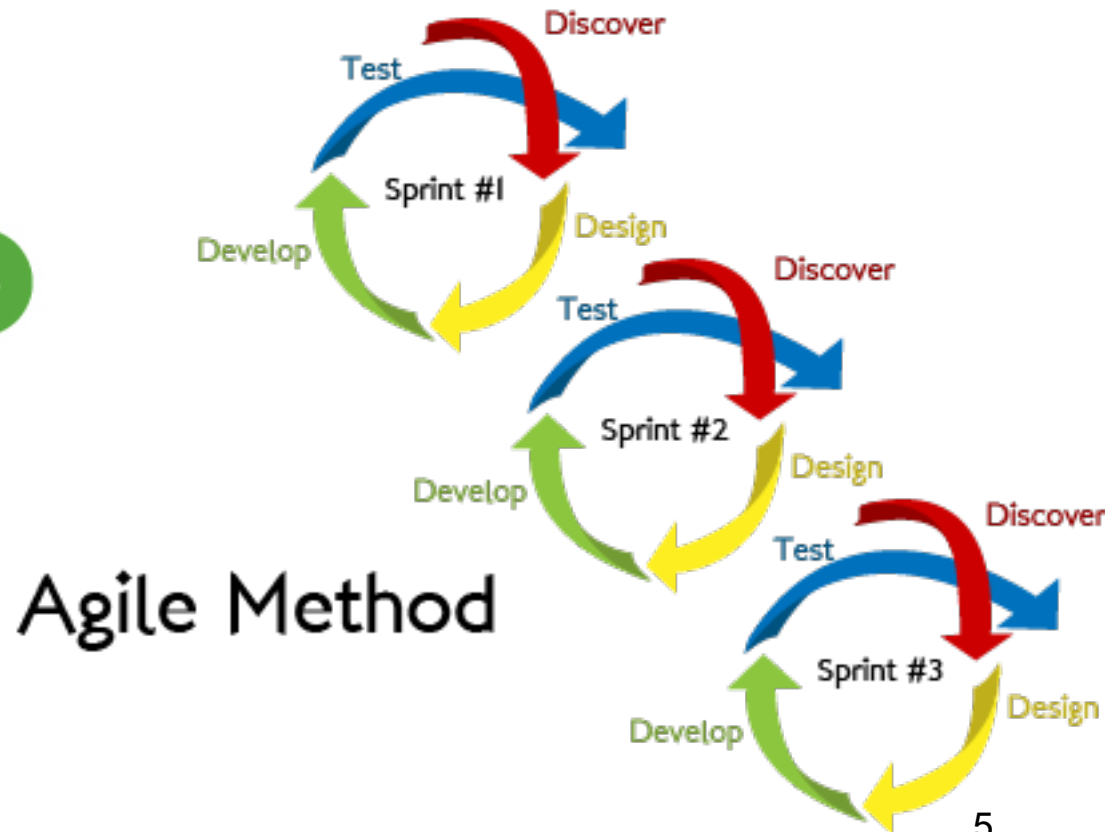


As-is team



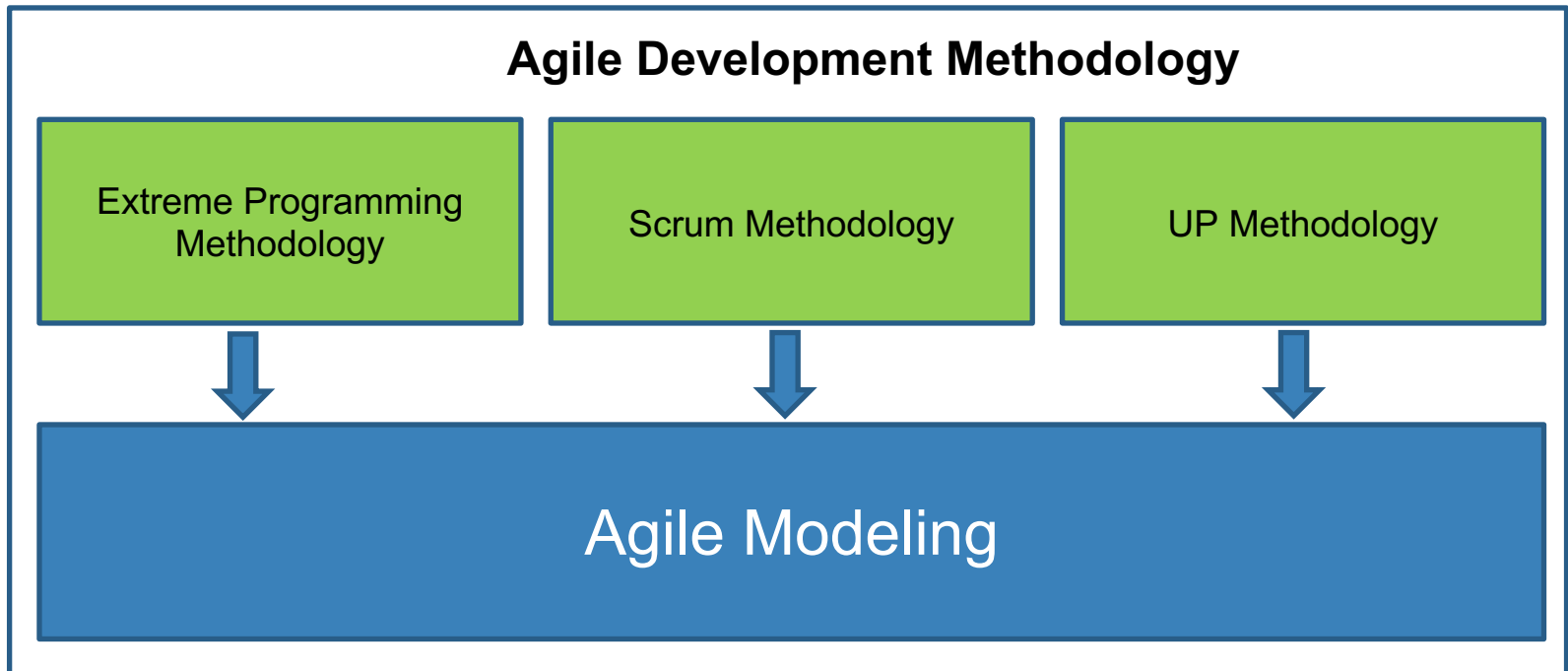


ปัญหาของระบบเดิม ?



Adaptive Development

- ▷ Agile Methodology
 - Extreme Programming
 - Scrum



Agile Methodology

- ▷ เป็นวิธีการพัฒนาระบบที่เน้นความรวดเร็ว และคล่องตัว(Rapid System Development)
- ▷ มุ่งเน้นความสำคัญที่ตัวระบบที่จะพัฒนามากกว่ากระบวนการออกแบบ และการใช้เอกสาร เพื่ออำนวยความสะดวกให้ทีมพัฒนาสามารถพัฒนาระบบได้รวดเร็วยิ่งขึ้น
- ▷ วิธีการนี้ยังตั้งอยู่บนพื้นฐานของวงจรการพัฒนาระบบ (System Development Life Cycle)
- ▷ ไม่สามารถแยกขั้นตอนการศึกษาความต้องการระบบ การออกแบบระบบ การพัฒนาโปรแกรมตามแบบ การทดสอบ และการนำไปใช้ออกจากกัน อย่างเด็ดขาด

ปรัชญาและคุณค่าของของแนวทางการพัฒนาซอฟต์แวร์แบบ Agile (Agile Manifesto)

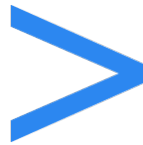
1. เน้นความพอใจของลูกค้า มีการส่งมอบงานอย่างต่อเนื่อง
2. ยอมรับการเปลี่ยนแปลงความต้องการของลูกค้า
3. มีการส่งมอบงานบ่อยๆทุก 2 สัปดาห์
4. ลูกค้า และทีมพัฒนาต้องทำงานร่วมกันทุกวันตลอดโครงการ
(ทีมพัฒนาไปทำงานที่ Site ลูกค้า)
5. ทีมพัฒนามีอำนาจการตัดสินใจกันเองได้ ไว้วางใจซึ่งกันและกัน
และต้องมีความรับผิดชอบ

ปรัชญาและคุณค่าของของแนวทางการพัฒนาซอฟต์แวร์แบบ Agile (Agile Manifesto)

6. การติดต่อ ประสานงานกันต้องพูดคุยด้วยตนเอง
7. วัดความก้าวหน้าของงานที่ตัวซอฟต์แวร์
8. กระบวนการทำงาน ให้ทำไปเรื่อยๆ ค่อยๆทำ ส่งมอบงานทีละงาน
9. ทีมพัฒนามีความสนใจเทคนิคต่างๆ มีการแลกเปลี่ยนซึ่งกันและกัน
10. เน้นการออกแบบง่ายๆ ไม่ซับซ้อน ทำให้แก้ไขง่ายเมื่อพบความเปลี่ยนแปลง
11. ทีมพัฒนา มีความรับผิดชอบในหน้าที่ของตนเอง
12. ทีมพัฒนา และลูกค้าต้องมีการทำงานคงที่ไปเรื่อยๆ ไม่เร็วหรือช้าเกินไป

Core Value of Agile

- ▷ Individuals interactions
- ▷ Working software
- ▷ Customer collaboration
- ▷ Responding to change



- ▷ Process and Tools
- ▷ Comprehensive documentation
- ▷ Contract negotiation
- ▷ Following a plan

- ❑ คนและการสื่อสารมีความสำคัญมากที่สุด
- ❑ สามารถเลี้ยวซ้าย เลี้ยวขวา เปลี่ยนแปลงได้ตลอดเวลา
- ❑ การวางแผนสำคัญกว่าการทำตามแผน

Extreme Programming

Extreme Programming คือกฎระเบียบการพัฒนาซอฟต์แวร์ โดยจะเน้นความสนใจไปที่สมาชิกในทีมทั้งหมด

- ▷ กำหนดวิธีการที่จะทำให้ลูกค้าและโปรแกรมเมอร์ทำงานร่วมกันในทีม
- ▷ สามารถตอบสนองต่อการเปลี่ยนแปลงของผู้ใช้ได้
- ▷ ตรวจพบข้อผิดพลาดได้เร็วกว่าวิธีอื่นๆ เนื่องจากมีการทดสอบ (Unit Testing) ตั้งแต่แรกเริ่ม
- ▷ เหมาะสำหรับทีมงานที่มีขนาดเล็กจนถึงขนาดกลางที่มีสมาชิกไม่เกิน 15 คน
- ▷ แบ่งการพัฒนาออกเป็นกลุ่มของงานย่อย โดยแต่ละงานย่อย ควรจะสำเร็จภายในระยะเวลาสั้น ๆ
- ▷ ควรมีการพบปะระหว่างทีมพัฒนาในแต่ละวัน

XP : 12 Practices (1)

○การวางแผน Planning Game

○ส่งมอบกับลูกค้าบ่อย Small Release

○ตั้งชื่อให้เข้าใจง่าย สื่อความหมายชัดเจน System Metaphor

○การใช้วิธีการต่างๆหรือแม้ ฟังก์ชันการทำงานที่เรียบง่ายเพื่อง่าย

ต่อการเข้าใจ Simple Design

XP : 12 Practices (2)

- การทดสอบก่อนเขียนโค้ด Testing Driven Development
- ทำให้ระบบเข้าใจง่ายที่สุด Design improvement
- ทำงานเป็นคู่ Pair Programming
- การอนุญาต ให้ผู้อื่นสามารถแก้ไขในส่วนของตนเองได้
Collective code ownership

XP : 12 Practices (3)

- รวมโค้ดเมื่อการแก้ไขเสร็จ Continuous integration
- ทำงานไม่ควรเกิน 40 ชม ต่อ สัปดาห์ Forty-hour week
- ทุกคนมีส่วนร่วม Whole Team
- พัฒนาโค้ดให้เป็นมาตรฐาน Coding Standard

XP : Pair Programming

- ▶ The basic principle is simple, every line of code is written while two programmers are in front of the screen.



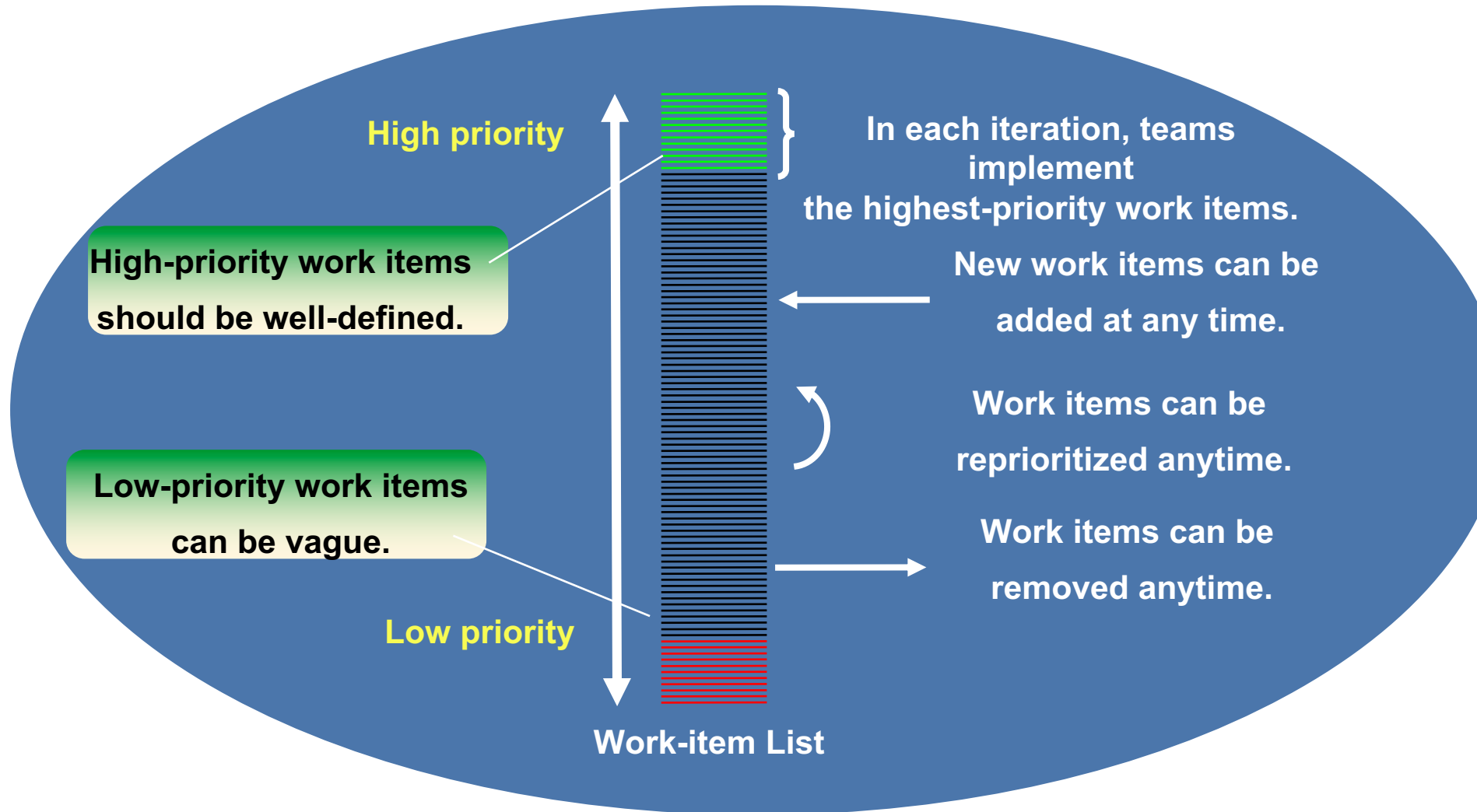
SCRUM

- ▷ Scrum เริ่มต้นจากการศึกษาในปี 1986 โดยนายทาเคอูจิ และ นายโนนากะ
- ▷ โดยเป็นการศึกษาเกี่ยวกับการบริหารจัดการระบบที่ใช้ทรัพยากรต่างๆจำนวนน้อยให้เกิดประโยชน์ที่สูงที่สุด
- ▷ ต่อมาทำการจัดรูปแบบของ Scrum ในอยู่ในรูปแบบที่เป็นมาตรฐานที่ได้รับการยอมรับกันในอุตสาหกรรมซอฟต์แวร์ โดยออกตีพิมพ์บน OOPSLA ในปี 1995

SCRUM

- ▷ หลักการพื้นฐานของ Scrum นั้นมาจาก Iteration และ Incremental เช่นเดียวกับ methodology ทุกตัวของ Agile
- ▷ เพื่อให้สามารถส่งมอบ software ได้เร็วๆ ตาม priority ของลูกค้า
- ▷ ซึ่ง scrum นั้นจะเด่นในเรื่องของ process และระยะเวลาในการทำงาน
- ▷ time-box ที่ชัดเจน รวมทั้งบทบาทหน้าที่(Role)ต่างๆ ก็ชัดเจนด้วย
- ▷ Scrum เป็นเฟรมเวิร์คที่เรียบง่ายที่ใช้ในการจัดการทีมงานที่มีอยู่ให้สามารถทำงานสำเร็จได้มากขึ้น และมีประสิทธิภาพสูง

Prioritize



Scrum vocabulary

Term	Description
Sprint	A <30 day focused effort moving the team toward fixed goals
Product Backlog	A constantly prioritized to-do list
Sprint Backlog	A list of the highest priority items from the product backlog - planned to be completed in the current Sprint
Scrum Master	The coach for the product management team, works to ensure realization of sprint goals
Product Owner	Represents the customer and is responsible for prioritizing the backlog
Scrum Team	A group of 5-9 people who self-organize and have joint responsibility for the completed tasks

Scrum Roles

- ▷ **Product Owner** มีหน้าที่รับผิดชอบต่อไปนี้
 - กำหนดความสามารถและฟังก์ชันต่างๆที่ต้องการให้มีในระบบ
 - ระบุกำหนดการส่งมอบงาน
 - เป็นคนประเมินประโยชน์ที่ได้รับและผลตอบแทนในการลงทุน (Return On Investment : ROI)
 - จัดลำดับความสำคัญของงานแต่ละงาน
 - มีการปรับเปลี่ยนแผนงานตามความต้องการที่เปลี่ยนแปลงไป
 - ทำการตรวจสอบเพื่อรับ หรือปฏิเสธผลงานที่ได้มา
- ส่วนเจ้าของงาน หรือ ลูกค้า (Product Owner) จะต้องเป็นผู้นำทีม เป็นสมาชิกในทีม Scrum และใช้ความต้องการ (requirement) ของตนเองเป็นตัวผลักดันให้ทีมสร้างผลงานออกมา ตามลำดับความสำคัญที่ Product Owner กำหนดให้ ซึ่งจะตรงกับหลักของวิธีการ agile ที่ว่า “Business people and developers must work together daily through the project”

Scrum Roles

▷ Scrum Master มีหน้าที่รับผิดชอบดังต่อไปนี้

- ทำการจัดการให้ทีมทำงานได้เต็มประสิทธิภาพและได้ผลประโยชน์สูงสุด
- ทำหน้าที่ประสานงานระหว่างกลุ่ม
- จัดการกับปัจจัยภายนอกที่จะมากกระทบกับการทำงานของทีม
- จัดการให้เกิดกิจกรรมต่างๆที่เป็นส่วนประกอบของสกรัม เช่น การออก “Scrum Daily Meeting” , “Sprint Review” และ “Sprint Plan Meeting”

○ ใน Scrum บอกว่า ไม่สนใจว่าแต่ละคนตำแหน่งอะไร ทุกคนคือทีมเดียวกัน สามารถออกความเห็นได้ทุกเรื่อง และทั้งทีมมีหน้าที่ที่จะต้องช่วยกันพัฒนา software ให้สำเร็จให้ได้ อะไรที่เป็นปัญหาก็ช่วยกันรุมทำให้เสร็จ

Scrum

- ▷ Scrum Master ต้องคอยให้คำแนะนำกับทีมพัฒนา และ Product Owner เพื่อบอกว่า Product Owner จะต้องทำอะไรบ้าง และต้องทำอะไร เช่น Product Owner ต้องเขียน requirement ยังไงจึงจะสื่อสารได้อย่างถูกต้อง ครบถ้วน หรือ คอยช่วยกำหนด priority ของ requirement เป็นต้น
- ▷ นอกจากนั้นยังต้องคอยกำกับไม่ให้ Product Owner ทำเกินหน้าที่ด้วย เช่น ไม่ไปก้าวล่วงเรื่องเวลาเข้างานหรือเลิกงานของทีมพัฒนา เป็นต้น คอยดูแลให้แต่ละฝ่ายทำหน้าที่ของตน และทำยังไงให้ The Team และ Product Owner ทำงานด้วยกันได้นั่นเอง

Scrum

- ▷ โดยปกติองค์กรที่นำวิธีการ Scrum มาใช้ Project Manager จะทำหน้าที่เป็น Scrum Master
- ▷ สิ่งที่พึงระวัง คือ โดยปกติ Project Manager จะพยายามเร่ง หรือสั่งงานทีมพัฒนา และยังเป็นผู้กำหนดงานต่างๆ ให้กับสมาชิกในทีม แต่วิธีการ Scrum ไม่ทำงานในลักษณะนั้น
- ▷ หน้าที่ของ Scrum Master จะต้องคอยกระตุ้น หรือสร้างกำลังใจให้ทีมพัฒนา ทำงานได้ด้วยตัวเองมากกว่าจะมาบังคับกัน
- ▷ ส่วนทีมพัฒนาไม่มีคนคอยบอกว่าใครต้องทำอะไร สมาชิกในทีมก็ต้องกระตือรือร้นมากกว่าเดิม ซึ่งถ้าทีมพัฒนา ไม่มีวุฒิภาวะมากพอ Agile จะสำเร็จได้ยาก ถือเป็นจุดอ่อนของวิธีการ Agile เพราะวิธีการ Agile ไม่ได้ใช้ process หรือ tool มาบังคับให้คนทำงาน แต่ใช้สำนึกและความรับผิดชอบของคนทำงานแทน

Scrum Roles

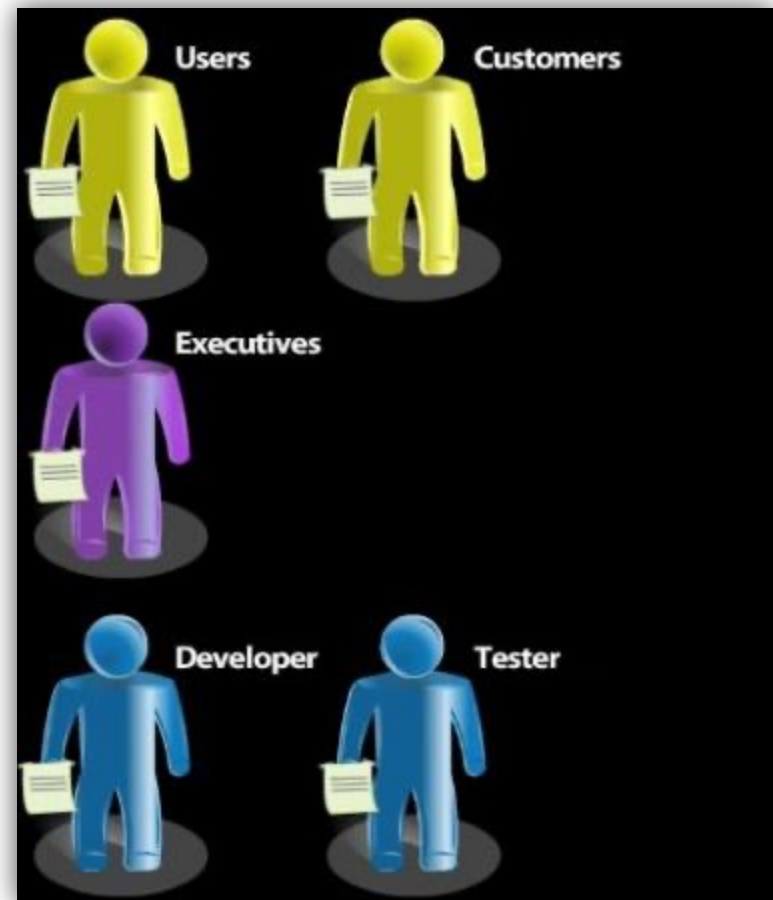
▷ Team: ในทีมควรมีทุกตำแหน่งที่จำเป็นครบถ้วน ทั้ง System Analyst, Programmer, Tester, etc. โดยไม่มีการแยกออกไปเป็นอีกทีมหนึ่ง

เป็นการทำงานแบบ Cross Function ซึ่งความรับผิดชอบของทีมเป็นดังต่อไปนี้

- ทำการเลือก Sprint Goal ที่ต้องการและทำงานตามเป้าหมายนั้นเพื่อให้ได้ผลตามที่กำหนดเอาไว้
- จัดระบบการทำงานให้มีประสิทธิภาพ
- ช่วยกันพัฒนางาน และแก้ปัญหาเพื่อให้งานเสร็จลุล่วง
- ทำการนำเสนองานที่เสร็จแล้วให้กับทางเจ้าของงานได้รับทราบ

Scrum

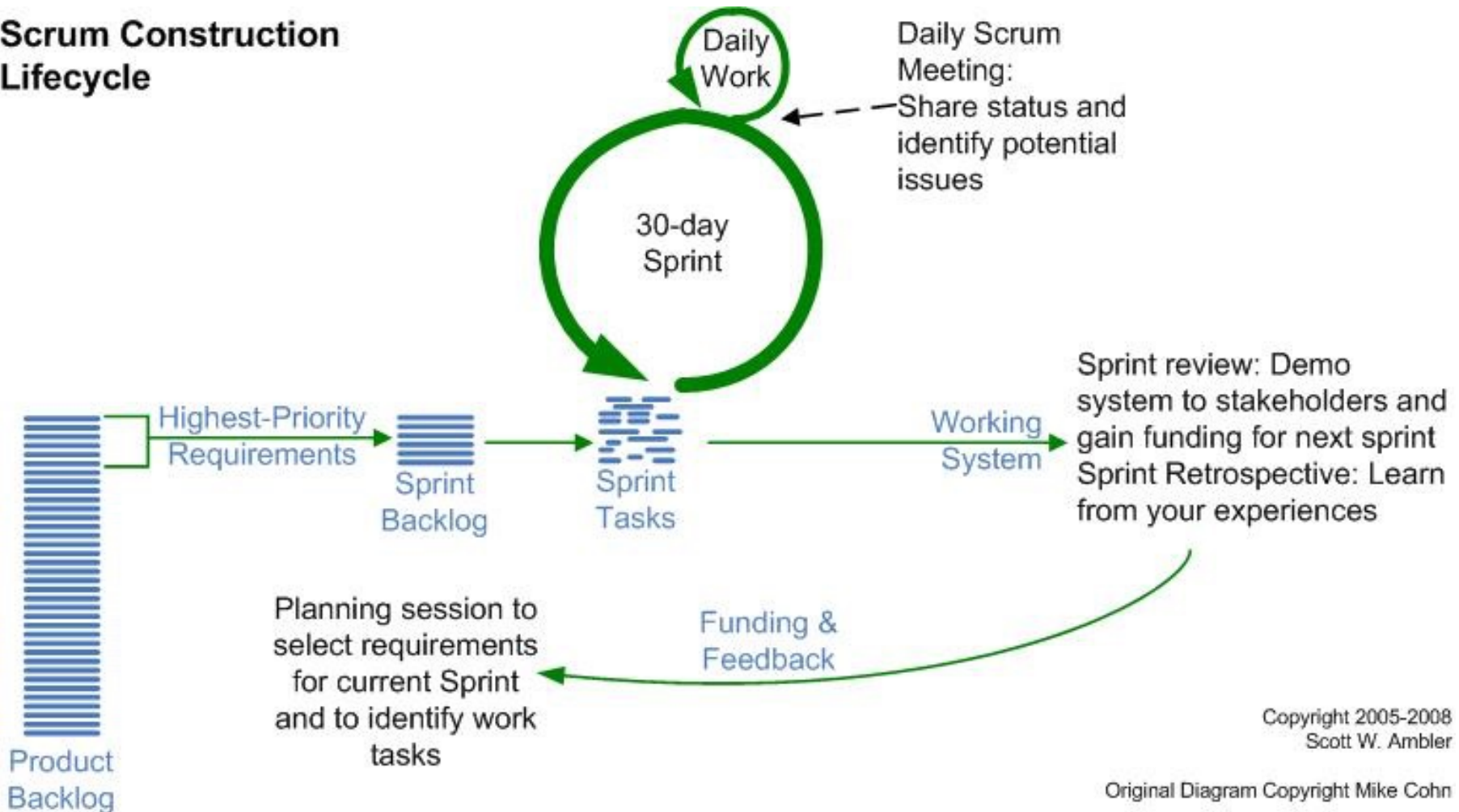
- ทีมงาน (Role)
 - Scrum Team
 - Scrum Master
 - Product Owner
- วิธีการทำงาน (Process)
 - Sprint
 - Backlog
 - Dairy Scrum



Agile example: The scrum process

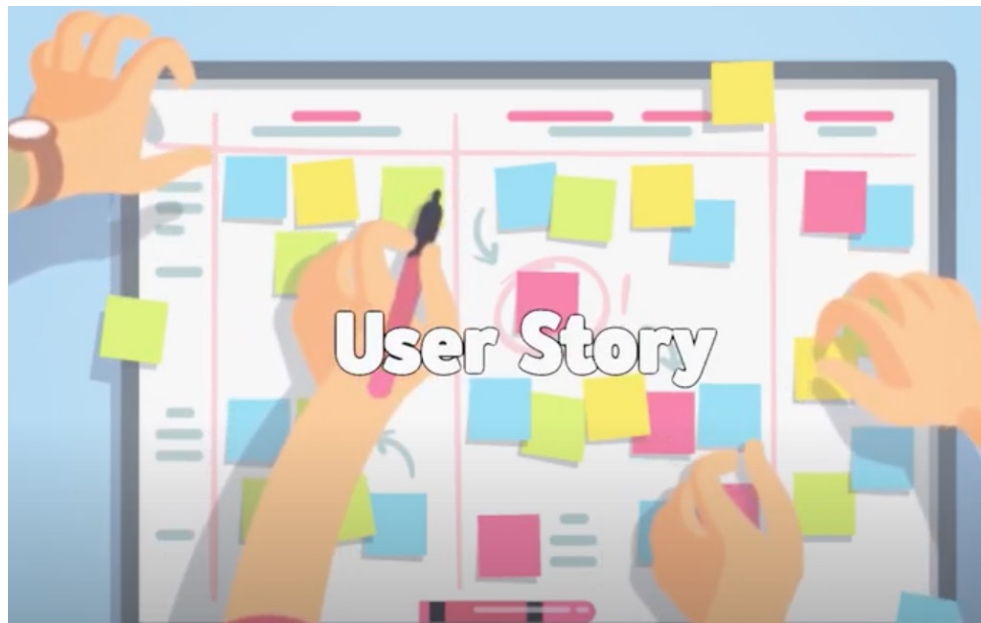
Step 1: Create the product Backlog


Scrum Construction Lifecycle




Product Backlog

- ▶ กระบวนการนี้เริ่มต้นที่เจ้าของงานที่จะเป็นผู้กำหนดและเรียงลำดับความสำคัญความต้องการและสิ่งที่ต้องการประกอบกันขึ้นมาเป็นระบบงาน สิ่งที่ได้จากกระบวนการนี้ถูกเรียกว่า Product Backlog





#	Backlog Item (User Story)	Story Point
1	As a Teller I want to be able to find clients by last name, so that I can find their profile faster	4
2	As a System Admin I want to be able to configure user settings so that I can control access	2
3	As a System Administrator I want to be able to add new users when required so that...	2
4	As a data entry clerk, I want the system to automatically check my spelling so that...	1



New Customer Touch Points – Financial Report System

- | | | | |
|----------------------|----------------------|----------------------|------------------------------|
| 1. ล็อกอินระบบ 1 | 5. ดึงข้อมูลรายงาน 2 | 9. ปรี้นทีใบจ่ายเงิน | 13. ดึงข้อมูลรายงาน 3 |
| 2. ดึงข้อมูลรายงาน 1 | 6. เซฟไฟล์2ลงเครื่อง | 10. หัวหน้าเซ็น | 14. เซฟไฟล์3ลงเครื่อง |
| 3. เซฟไฟล์1ลงเครื่อง | 7. อัปโหลดสองไฟล์ | 11. ส่งใบจ่ายเงิน | 15. อัปโหลดไฟล์3 |
| 4. ล็อกอินระบบ 2 | 8. กดปุ่มเทียบรายงาน | 12. ล็อกอินระบบ 3 | 16. กดปุ่มเทียบรายงาน |
| | | | 17. โทรียืนยันกับทีมซัพพอร์ท |
| | | | 18. ปรี้นทีใบปรับยอดเงิน |
| | | | 19. ส่งใบปรับยอดเงิน |

New Estimates of Effort Remaining at end of Day...									
Product Backlog Item	Sprint Task	Volunteer	Initial Estimate of Effort	1	2	3	4	5	6
As a buyer, I want to place a book in a shopping cart	modify database	Sanjay	5	4	3	0	0	0	
	create webpage (UI)	Jing	3	3	3	2	0	0	
	create webpage (Javascript logic)	Tracy & Sam	2	2	2	2	1	0	
	write automated acceptance tests	Sarah	5	5	5	5	5	0	
	update buyer help webpage	Sanjay & Jing	3	3	3	3	3	0	
	...								
Improve transaction processing performance	merge DCP code and complete layer-level tests		5	5	5	5	5	5	
	complete machine order for pRank		3	3	8	8	8	8	
	change DCP and reader to use pRank http API		5	5	5	5	5	5	
...						
Total			50	49	48	44	43	34	

User Story

- User story เป็นกระบวนการในการเก็บรวบรวมความต้องการของผู้ใช้งานในกระบวนการพัฒนาระบบแบบ Agile ที่พยายามนำจุดดีของการเก็บรวบรวมความต้องการ แบบที่เป็นทางการที่มีเอกสารครบถ้วนและแบบปากเปล่า ที่สามารถทำได้รวดเร็ว โดยมีขั้นตอนดังต่อไปนี้
 - การทำการ์ด (Card)
 - การสนทนา (Conversation)
 - การยืนยัน (Confirm)



Understanding the basic differences between Epic, Story, and Task



Epic

Story

Tasks

New Year Party

Shopping for the party

Decorating the house

Cooking for the Guests

Raw materials for cooking

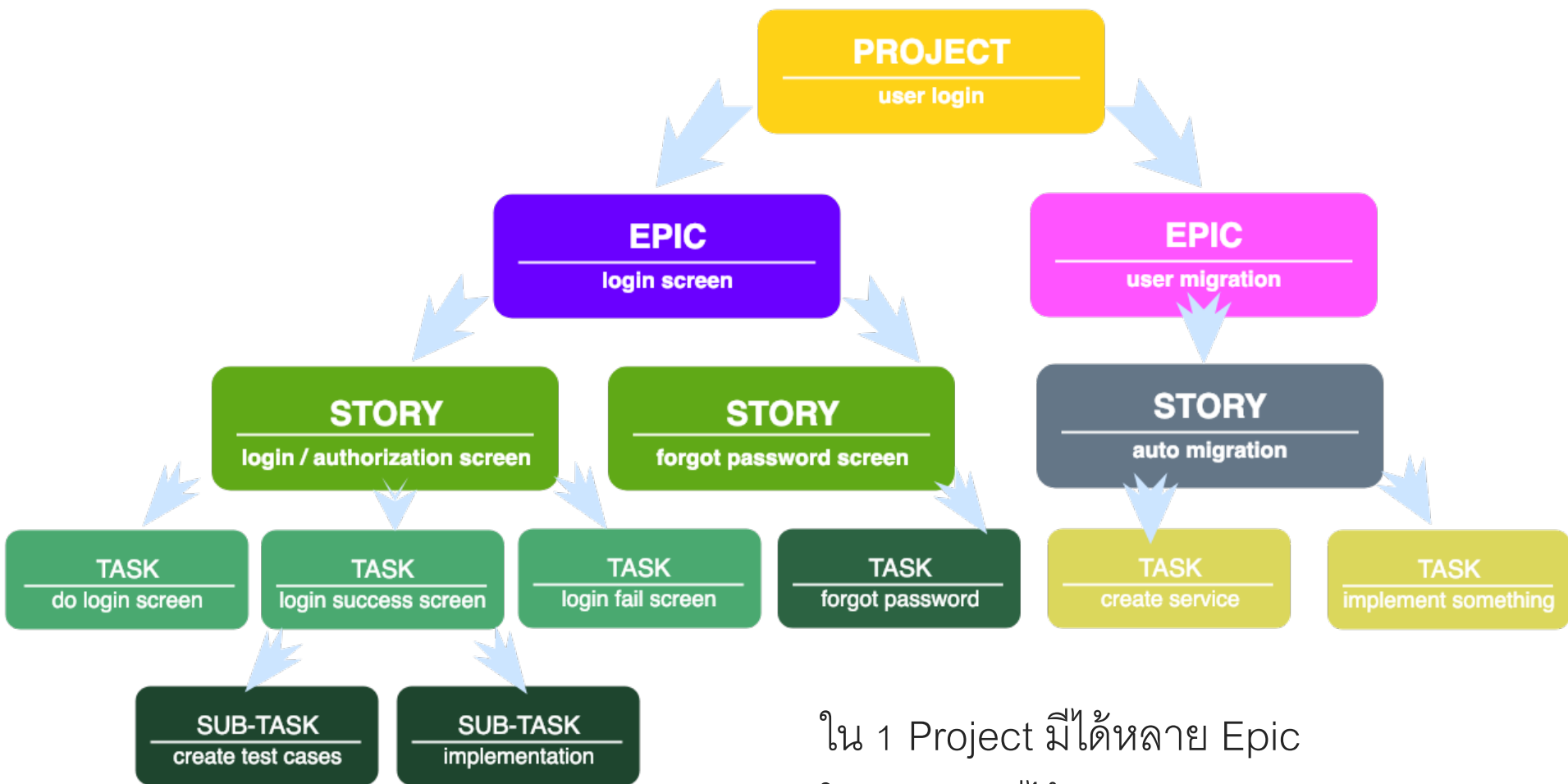
Gifts for the guests

Painting the exterior area

Put up a gift tree

Set-up the kitchen clean

Cut the green vegetables



ใน 1 Project มีได้หลาย Epic

ใน 1 Epic มีได้หลาย Story

ใน 1 Story มีได้หลาย Task

ใน 1 Task มีได้หลาย Sub-Task

Stories	Not started	In progress	Done
<div data-bbox="85 329 440 578"><p>Story #1</p><hr/><hr/><hr/></div>			<div data-bbox="1450 358 1632 486">Task A</div> <div data-bbox="1655 358 1837 486">Task B</div> <div data-bbox="1450 506 1632 635">Task C</div>
<div data-bbox="85 681 440 929"><p>Story #2</p><hr/><hr/><hr/></div>	<div data-bbox="504 681 685 809">Task A</div>	<div data-bbox="1051 681 1232 809">Task C</div>	<div data-bbox="1470 681 1651 809">Task B</div>
<div data-bbox="65 1029 413 1278"><p>Story #3</p><hr/><hr/><hr/></div> <div data-bbox="85 1296 162 1370"></div>	<div data-bbox="504 1029 685 1158">Task B</div> <div data-bbox="724 1029 906 1158">Task D</div> <div data-bbox="504 1198 685 1326">Task A</div> <div data-bbox="724 1198 906 1326">Task C</div>		

User Story

EPIC	EPIC description	User Story	Size	Task	Status	Developer	Co-developer	Comments
User authorization	ระบบต้องรองรับ User role ซึ่งจะกำหนดสิทธิ์การเข้าใช้ functions ในระบบ							
		ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถสร้าง Function ได้เพื่อกำหนดสิ่งที่ User สามารถใช้ได้ในระบบ	S		todo			
		ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถค้นหา Function ตามกลุ่ม Function ที่มีอยู่ในระบบได้	S		todo			
		ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถที่จะสร้าง User roles ใหม่ได้เพื่อใช้ในการกำหนดสิทธิ์ User ว่า Functions ใดสามารถใช้ได้บ้าง	M		todo			1 Roles สามารถมีได้หลาย Functions
		ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถที่จะกำหนดค้นหา User roles โดย RoleId และ RoleName ได้เพื่อดูรายชื่อ User ทั้งหมดที่มี Role นี้	M		todo			
		ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถที่จะแก้ไขรายการ Functions ของ User role ได้เพื่อเพิ่มหรือลด Functions ของ User role นั้นๆ	S		todo			

K Travel project ☆

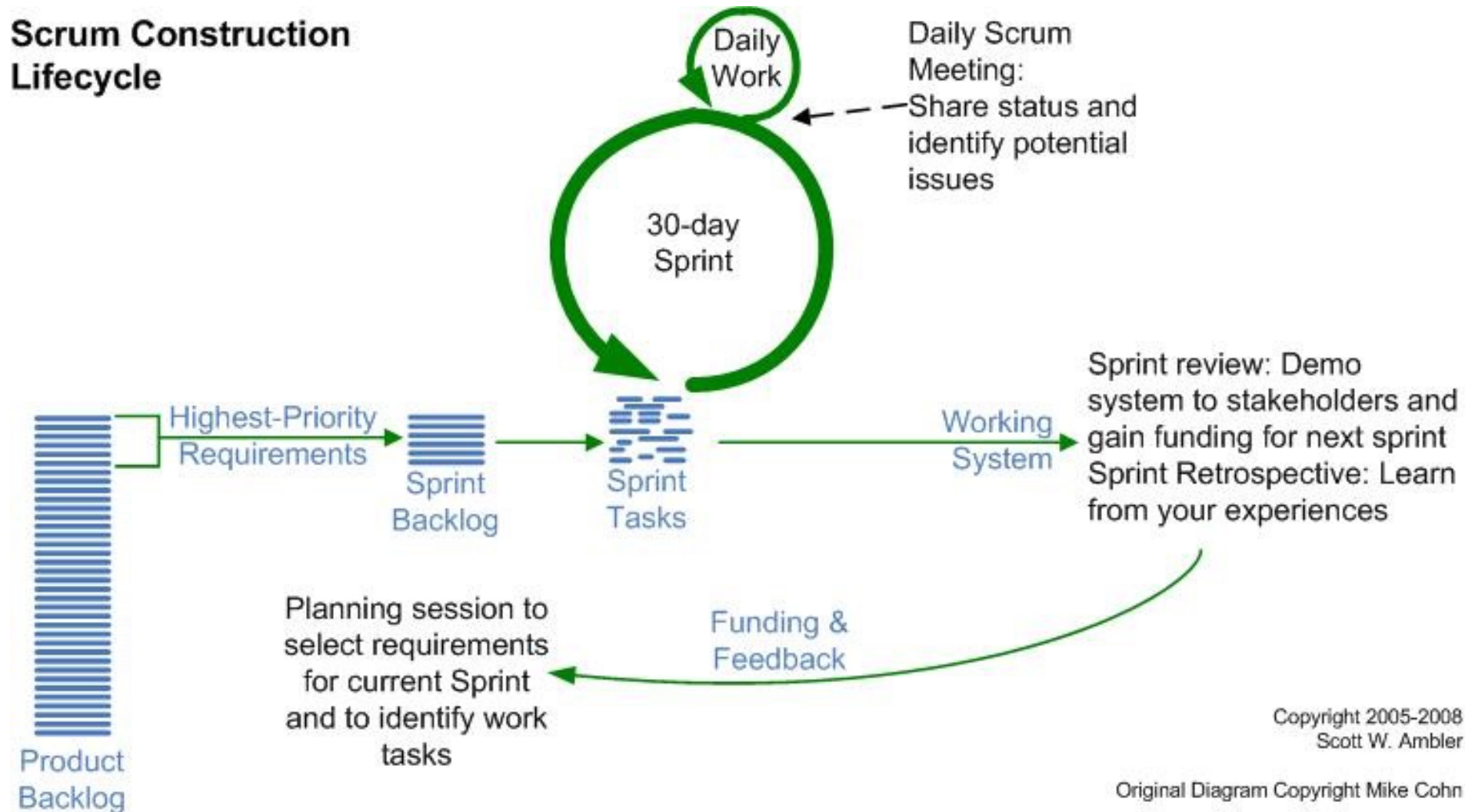
File Edit View Insert Format Data Tools Help All changes saved

A	B	C	D	E	F	G	H
EPIC-ID	EPIC	EPIC description	User Story	Size	Status	Team	Comments
1	User authorization	ระบบต้องรองรับ User role ซึ่งจะกำหนดสิทธิ์การเข้าใช้ functions ในระบบ				ทีม A	
			ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถที่จะกำหนด User role ได้	Large	todo		
			ในฐานะที่เป็นผู้ที่ไม่เคยใช้ระบบ ฉันสามารถที่จะ register ได้ เพื่อสามารถเข้าใช้ระบบ	Medium	todo		
					todo		
2	User authentication	ระบบต้องมีการตรวจสอบ username และ password ของ user ก่อนสามารถเข้าใช้ระบบได้					
			ในฐานะที่เป็นผู้ใช้ระบบ ฉันสามารถเข้าสู่ระบบโดย username and password	Small	todo		
			ในฐานะที่เป็นผู้ใช้ระบบ ฉันสามารถเปลี่ยน password ได้	Small	todo		
3	การจัดการ แพคเกจ ทีวี						
			ในฐานะที่เป็นผู้ควบคุมระบบ ฉันสามารถสร้างแพคเกจใหม่ได้	Medium	todo		

No	Requirement	User Story
1	Login	As a user, I want to create my own account so that I can login to use the system.
2	Login	As a user, I want a confirmation email after I create my account so that I can be sure about my privacy.
3	Login	As a user, I want the system to show how strength my password is, so that I can ensure my security.
4	Login	As a user, I want to be able to login with my ORBRS account, so that I can use the system.
5	Login	As a user, I want to be able to login with my facebook account, so that I can use the system.
6	Books	As an administrator, I want to add book information, so that the users can see it.
7	Books	As an administrator, I want to edit book information, so that I can keep it up-to-date.
8	Books	As an administrator, I want to delete a book, so that I can get rid of the outdated or invalid ones.
9	Books	As an administrator, I want the system to be integrated with Amazon, so that I can get information of newly-released books automatically.
10	Search Books	As a user, I want to search a book by its name, so that I can filter out irrelevant books.
11	Search Books	As a user, I want to search a book by its auther, so that I can filter out irrelevant books.
12	Search Books	As a user, I want to search a book by its published date, so that I can filter out irrelevant books.

Agile example: The scrum process

Step 2: Hold a sprint planning meeting

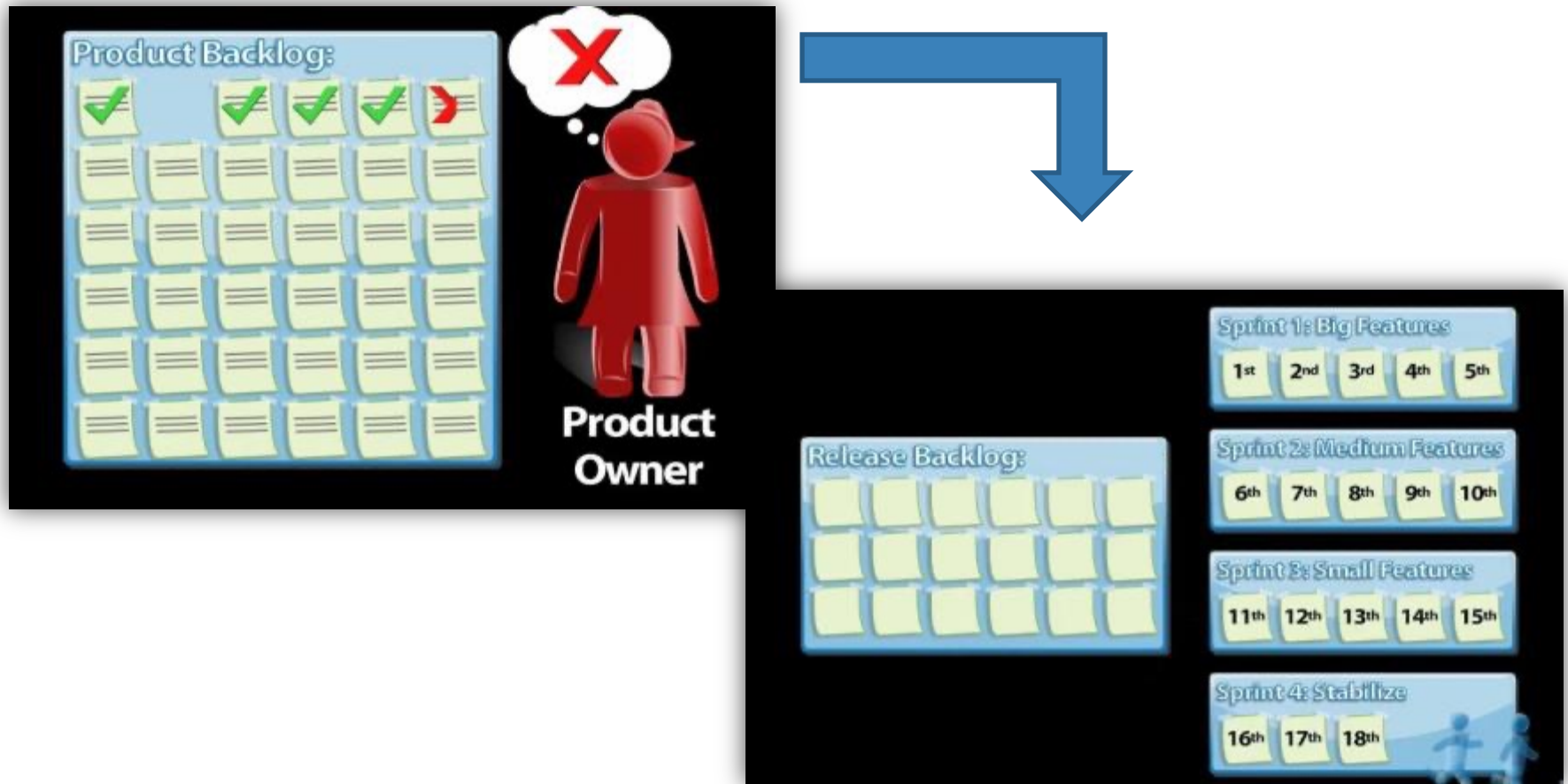


Sprint Planning Meeting

- ▷ หลังจากได้ Product Backlog ขั้นตอนถัดมาเป็นการนำ Product Backlog ที่ได้มาทำการรีวิวโดยทีมงาน Scrum เพื่อทำการประเมินความเป็นไปได้ รวมทั้งทรัพยากรที่ต้องใช้
- ▷ งานทั้งหมดจะทำการจัดสรรเป็นกลุ่มเล็กๆ ตามปัจจัย (ขนาดของทีม, ระยะเวลาที่เหลือ และความสามารถของทีมงาน) โดยสิ่งที่ได้หลังจากผ่านขั้นตอนนี้คือ Sprint Backlog

Scrum

▷ Release Backlog



ตัวอย่างการประเมิน
งานเทียบกับ
ทรัพยากรภายในทีม

Sprint Length	2 weeks
Workdays During Sprint	10 days

Team Member	Available Days During Sprint*	Available Hours Per Day	Total Available Hours During Sprint
Tracy	9 days	4 hours	36 hours [= 9 x 4]
Sanjay	10 days	5 hours	50 hours
Phillip	10 days	4 hours	40 hours
Jing	8 days	5 hours	40 hours

เฉพาะวันทำงานเท่านั้น

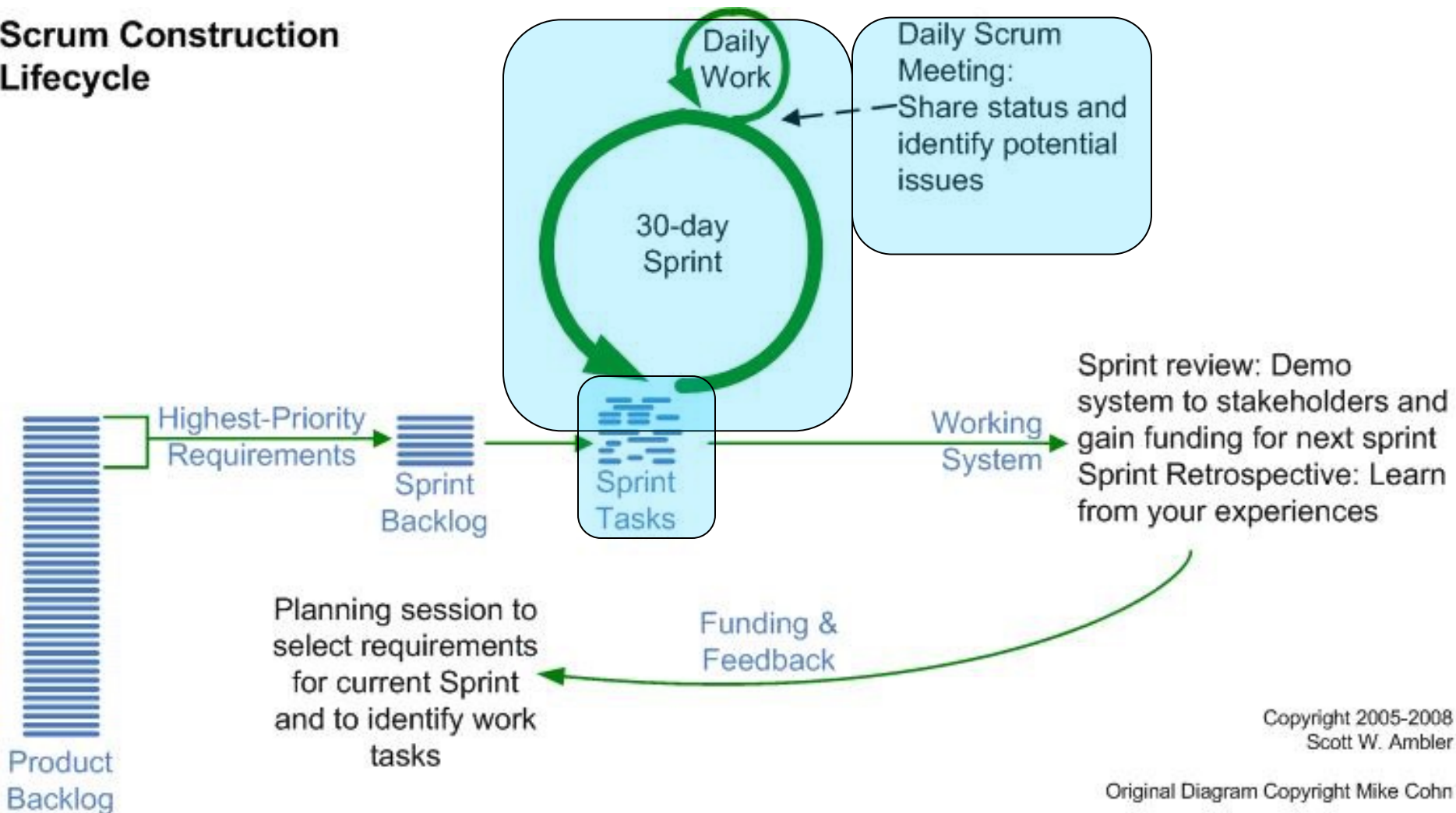
สปริ้นแบ็กล็อก

Backlog Item	Task	Owner	Initial Time Estimate
Enable all users to place book in shopping cart	Configure database and space IDs for Trac	Sanjay	4 hours
	Use test data to tune the learning and action model	Jing	2 hours
	Setup a cart server code to run as apache server	Philip	3 hours
	Implement pre-Login Handler	Tracy	3 hours
Upgrade transaction processing module (must be able to support 500 transactions /sec)	Merge DCP code and complete layer-level tests	Jing	5 hours
	Complete machine order for pRank	Jing	4 hours
	Change DCP and reader to use pRank http API	Tracy	3 hours

Agile example: The scrum process

Step 3: Complete the sprint

Scrum Construction Lifecycle



Copyright 2005-2008
Scott W. Ambler

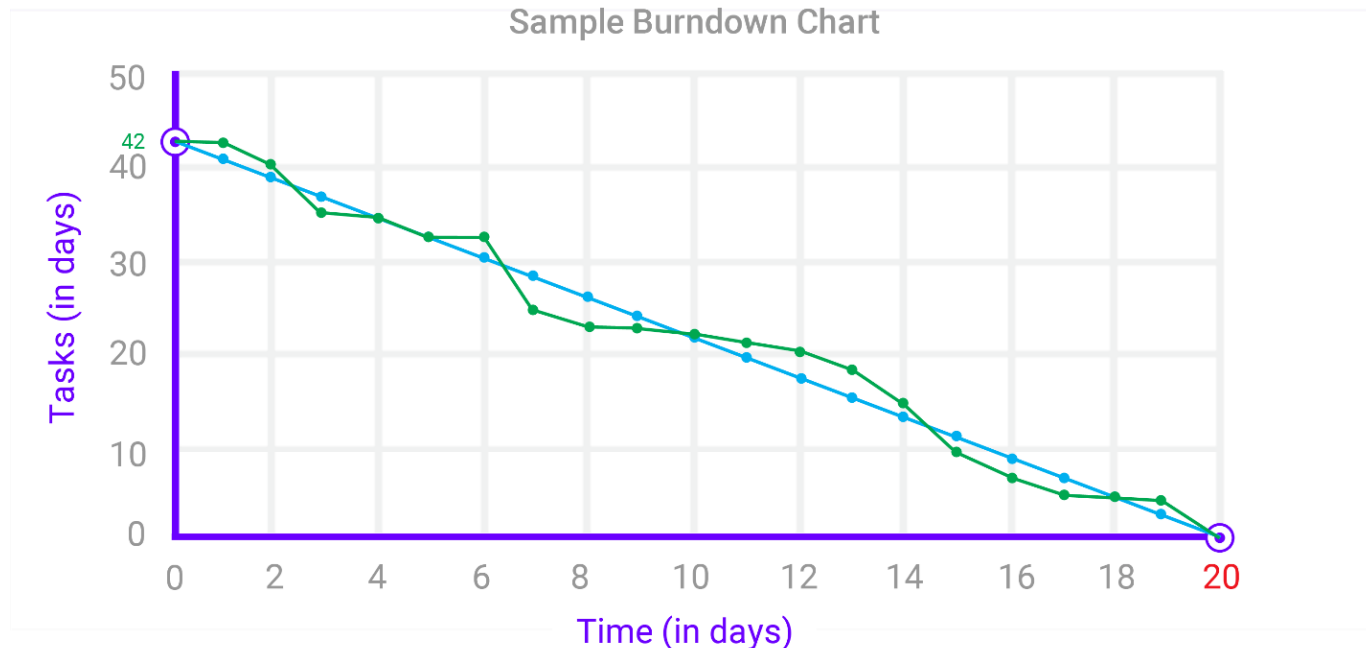
Original Diagram Copyright Mike Cohn

Daily Scrum Meeting

- ▷ การประชุมสกรัมประจำวัน(Daily Scrum Meeting)
- ▷ เป็นกิจกรรมที่ทำทุกวันโดยที่ สกรัมมาสเตอร์จะเป็นคนนำการประชุมนี้ โดยการประชุมจะใช้เวลาในการประชุมประมาณ 15 นาทีเพื่อเป็นการตรวจสอบว่าสถานะของสร์มในขณะนั้นเป็นอย่างไรบ้าง
- ▷ สำหรับทีมงานจะต้องตอบคำถาม 3 ข้อดังต่อไปนี้คือ
 - เมื่อวานทำอะไร (What did I do yesterday?)
 - วันนี้จะทำอะไร (What will I do today?)
 - มีอะไรเป็นอุปสรรคในการทำงานบ้าง (What impediments are in my way?)

Daily Scrum Meeting

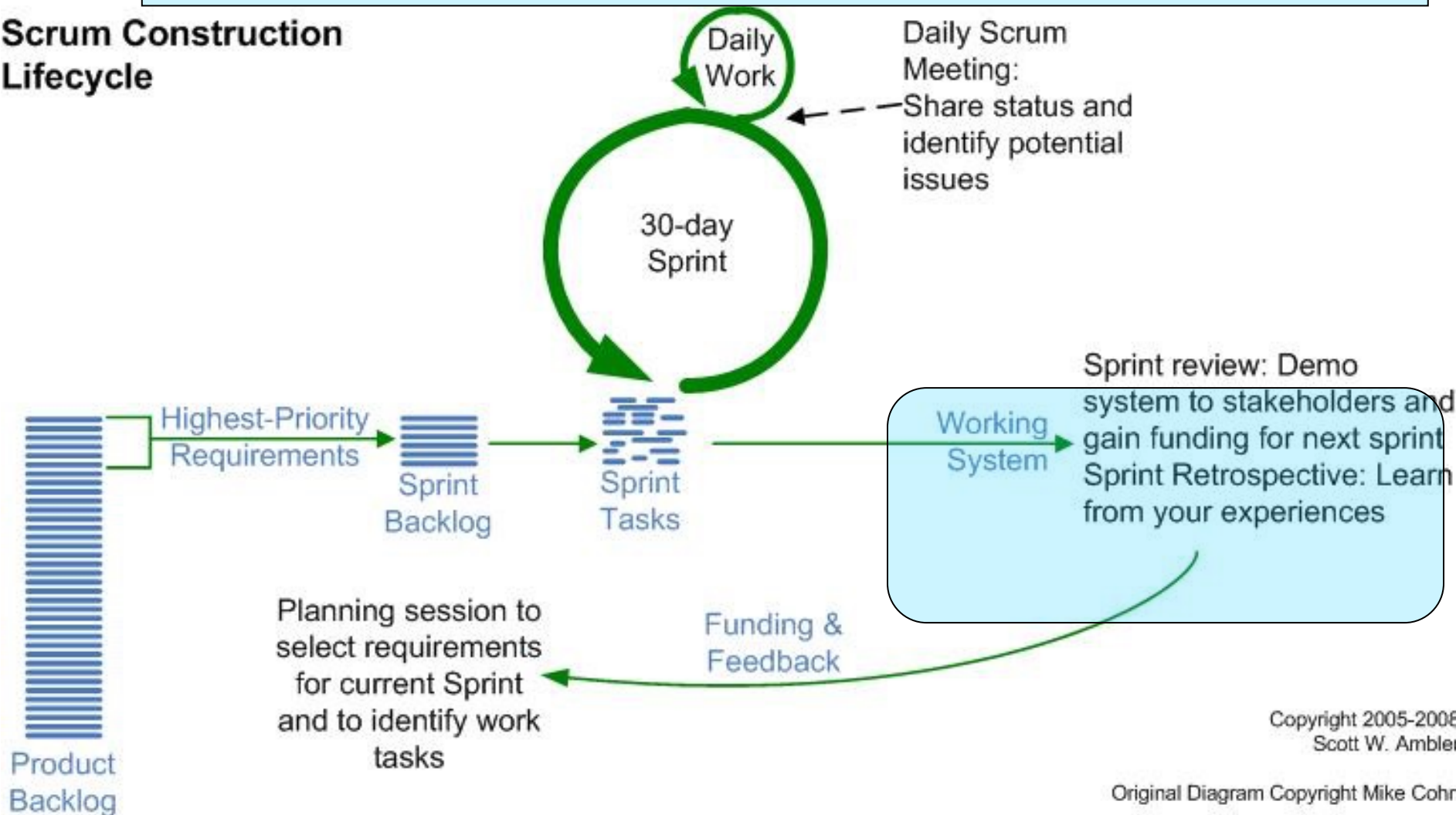
- ▷ สำหรับสกรัมมาสเตอร์มีหน้าที่ดังต่อไปนี้ในการประชุม
- ▷ ในการตรวจสอบว่ามีงานที่สิ้นสุด , งานที่กำลังทำ และ งานที่เพิ่มเติมขึ้นมา เพื่อดูว่ามีการเปลี่ยนแปลงมีผลกระทบอย่างไรโดยใช้ตัว เบิร์นดาวน์ชาร์ต (Burn down Chart) เป็นเครื่องมือที่ช่วยในการรวบรวมข้อมูล



Agile example: The scrum process

Step 4: Hold sprint review and sprint retrospective meetings

Scrum Construction Lifecycle



Copyright 2005-2008
Scott W. Ambler

Original Diagram Copyright Mike Cohn

วิธีการแบบ Agile ไม่ได้ใช้ process หรือ tool
มาบังคับให้คนทำงาน แต่ใช้สำนึกและความ
รับผิดชอบของคนทำงานแทน

▷ ข้อดีคือ:

1. รองรับการเปลี่ยนแปลง Requirement ได้ดี
2. สนุกกับการทำ Software
3. Fixed budget, fixed time และ Flexible feature
4. ลูกค้าเห็น product ได้ไว
5. มีความเสี่ยงน้อย โอกาส Project success สูง
6. เกิด Productivity มากกว่าการทำ Traditional process (waterfall)

▷ ข้อเสียคือ:

1. ทุกคนในทีมต้องเท่าเทียมกัน จะต้องไม่มี leader ที่ manage ทีม แต่ทุกคนในทีมต้องช่วยกัน
2. ไม่เหมาะกับงานหน่วยงานราชการ ที่เน้นทำเอกสารไว้ก่อน
3. ทุกคนในทีมต้อง Agile และต้องมีความเป็น Beginner mind คือเปิดรับฟังความเห็นจากคนอื่น ที่มีลักษณะ team ที่เรียกว่า Egoless team เท่านั้น
4. Agile อาจไม่เหมาะสำหรับบางคนและบางทีม แต่บอกได้ว่าคนไหน ทีมไหน เหมาะกับการทำ Agile

References

- ▷ <http://agilemanifesto.org/>
<http://www.allaboutagile.com/>
<http://www.agilenutshell.com/>

Thanks!

Any questions?