

Scrum Practices



Presented by Wiwat V.

Scrum Practices for Today

- Self-organizing Team (skip)
- Writing user story cards in Product Backlog (try)
- Estimation using Planning Poker (try)
- Writing Tasks in Sprint backlog (see sample)
- Burndown chart (see sample)
- Daily Scrum – Standup Meeting (Role Play)


เนื่องจากเวลาจำกัดแค่ 1 ชั่วโมง

Scrum in Brief



When it comes to AGILE!

Manifesto for Agile Software Development



We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

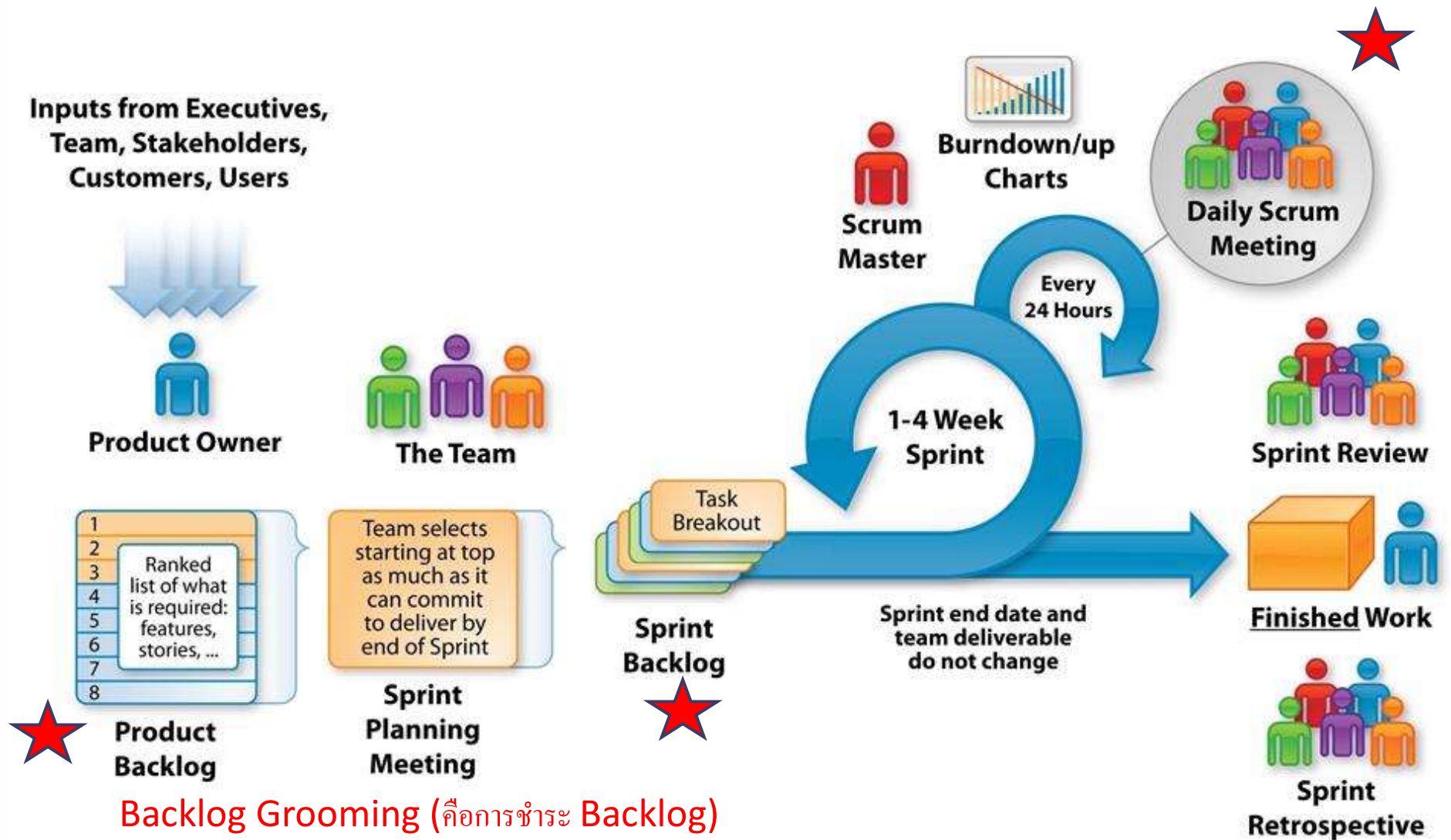
James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

12 Principle of Agile Software

- 1) Our highest priority is to satisfy the customer through **early and continuous delivery** of valuable software
- 2) **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage
- 3) **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter time scale
- 4) Business people and developers must **work together daily** throughout the project
- 5) Build projects around **motivated individuals**. Give them the environment and support they need, and trust them to get the job done
- 6) The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**
- 7) Working **software is the primary measure of progress**
- 8) Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to maintain a constant pace indefinitely
- 9) Continuous **attention to technical excellence and good design** enhances agility
- 10) **Simplicity**—the art of maximizing the amount of work not done—is essential
- 11) The best architectures, requirements, and designs emerge from **self-organizing teams**
- 12) At regular intervals, **the team reflects on how to become more effective**, then tunes and adjusts its behavior accordingly

The Agile - Scrum Framework



Important Vocab

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Let's Begin



Product Backlog Grooming Session

- Sometimes called “Time Story Session”
- Write user stories (it is possible to build a Product Backlog “from scratch” in a series of one or more Story Time sessions) User story = small requirement
- Break down user stories that are too big (epics)
- Improve user stories that are poorly written
- Estimate backlog items – using relative story points
- Add acceptance criteria
- Look deeper into the backlog to do longer-range technical planning (คุณภาพรวมของการใช้ technical)

Grooming = to care for one's appearance, to prepare for the position/purpose

User Story Structure/Format

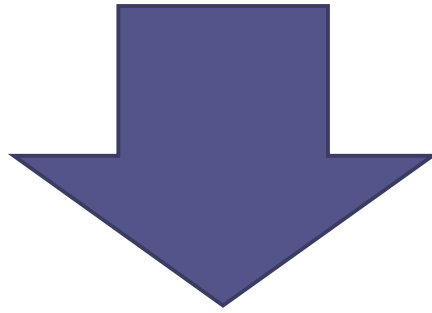
*As a <Type of User>,
I want <To Perform Some Task>,
So that <I can achieve some goal/benefit/value>.*

User Story Acceptance Criteria

Each User Story also has Acceptance Criterion defined, so that correctness of implementation of the user story is confirmed by passing the Acceptance Test that is based on the Acceptance Criterion.

What if the user story is too big?

- Too big means “can not directly used in the sprint”



EPIC

Scrum Epic

- Is a large/big user story
- An Epic should break down into stories that are probably small enough to implement directly
- Epic is typically used to group a set of user stories

Example of ATM system's Epics

Epic#1: Customer Services



Services is too big

As a customer, I want to get service for my bank account, so that I do not to wait in the line.

Epic#2: ATM Operator Services

As an ATM operator, I want to do daily operations for the ATM machine, so that the ATM would be ready for the next day service.

Epic#3: ATM Maintenance Services

As a Maintenance Engineer, I want to do the privilege operations for the ATM machine, so that the error would be fixed.

Epic#1: Customer Services

- User Story#1.1:
Let us take a look at how a user story is framed for the scenario of a Bank Customer **withdrawing** cash from ATM.
- User Story#1.2:
User would like to **transfer cash** from his/her bank accout to another bank account using ATM
- User Story#1.3:
.....**Pay Bill**.....

User Story#1.1: Customer's Cash Withdrawal

As a **Customer**,

I want to **withdraw cash from an ATM**,

So that **I don't have to wait in line at the Bank**

User Story#1.1: Customer's Cash Withdrawal

Acceptance Criterion 1:

Given that the account is creditworthy

And the card is valid

And the dispenser contains cash,

When the customer requests the cash

Then ensure the account is debited

And ensure cash is dispensed

And ensure the card is returned.

User Story#1.1: Customer's Cash Withdrawal

Acceptance Criterion 2:

Given that the account is overdrawn

And the card is valid

When the customer requests the cash

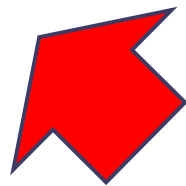
Then ensure the rejection message is displayed

And ensure cash is not dispensed

And ensure the card is returned.

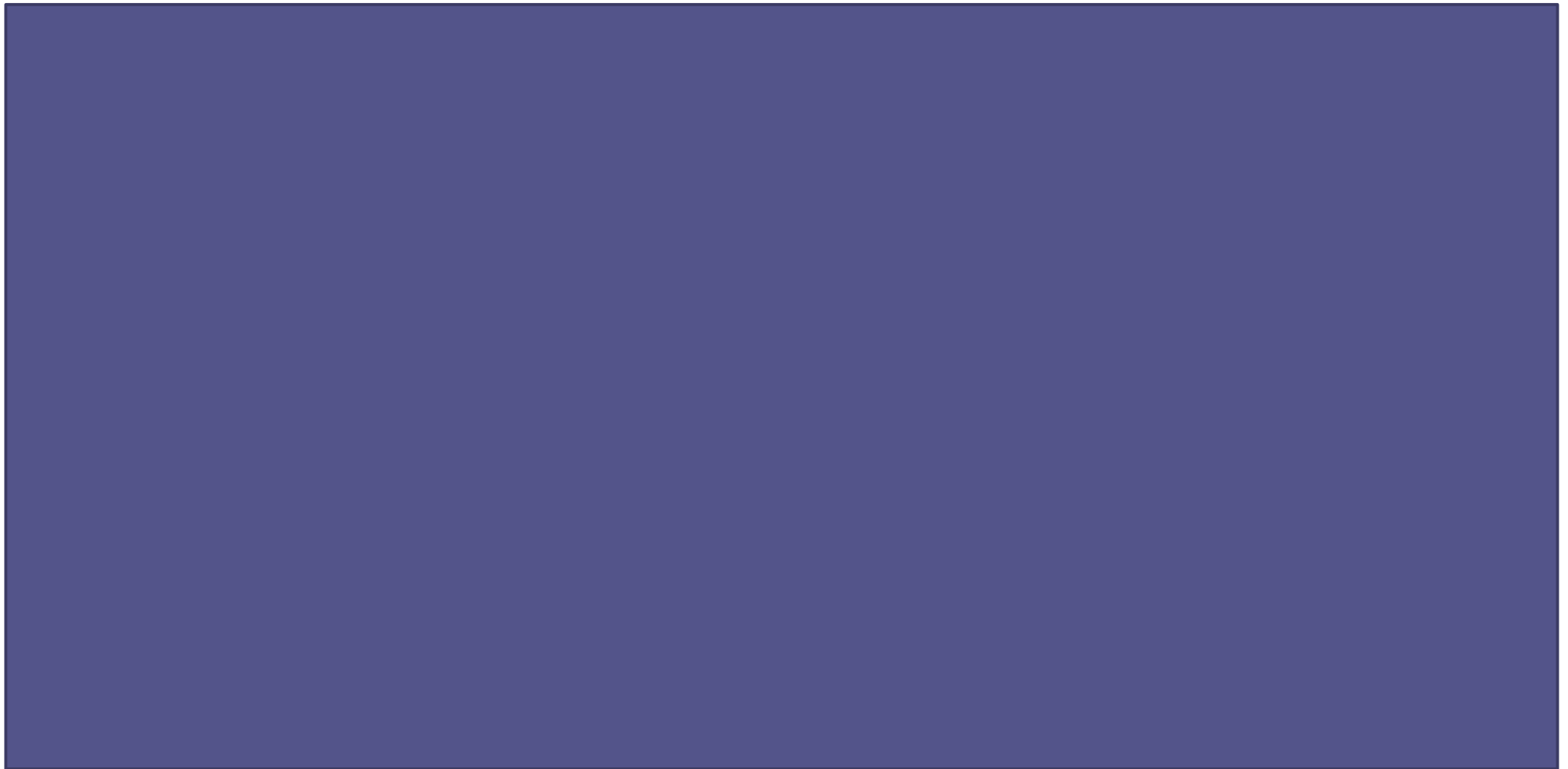
Exercise

- Who is responsible for writing user stories in product backlog?
- Who does the writer of the user stories?
- จงเขียน User Story และ Acceptance Criterion มา 1 ข้อ สำหรับการโอนเงินผ่านเครื่อง ATM
“User Story #1.2 Customer’s Cash Transfer” ใน EPIC#1 “



คำตอบ

User Story#1.2: Customer's Cash Transfer



Estimating the User Story

A series of horizontal lines in teal, light blue, and white, extending from the left edge of the slide and ending under the title.

Easy Exercise

“ใช้เวลาที่ในการเดินจากหน้าตึก 4 ไปยังตึกมหิตลาธิเบศร คณะบัญชี” (คนเดินด้วยความเร็วต่างกัน)

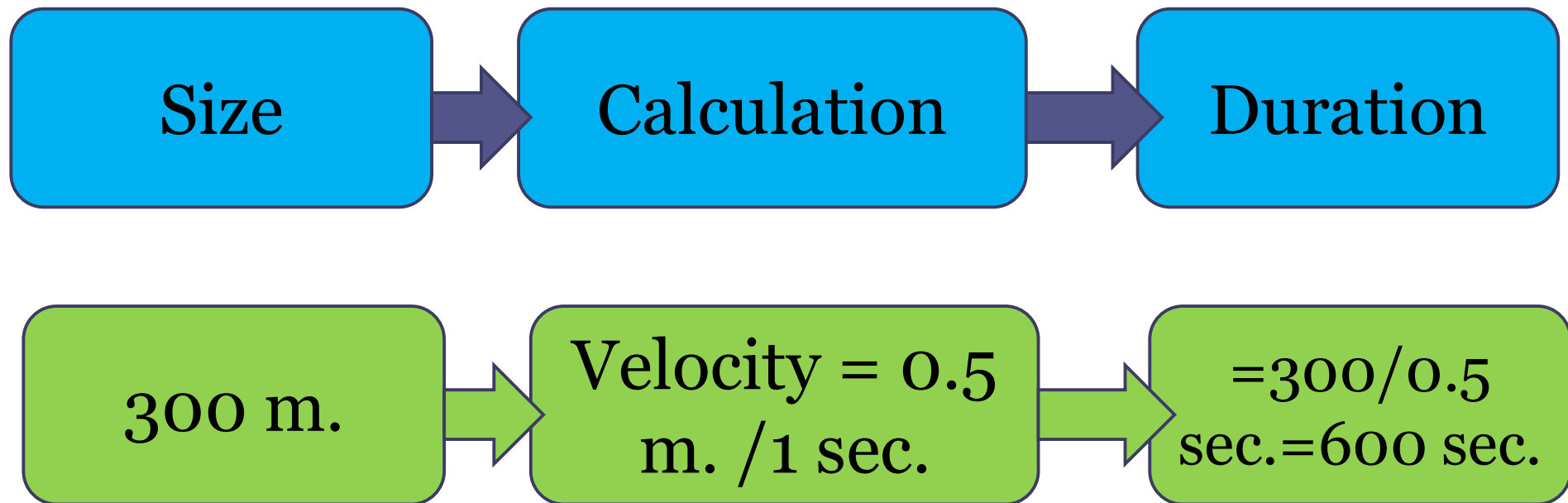
- ระยะทางจากหน้าตึก 4 ไปลานเกียร์ กำหนดเป็นค่า “2”
- ระยะทางจากบันไดหน้าตึก 4 ไปยังตึกมหิตลาธิเบศร คณะบัญชี จะมีค่าเป็นเท่าไรเมื่อเทียบกับ “2”

คำถามที่มักถูกถาม

- ซอฟต์แวร์จะเสร็จเมื่อไร
- ซอฟต์แวร์ต้องเสียเงินเท่าไร (หรือมีค่าใช้จ่ายเท่าไร)
- เรารู้คำตอบว่า
 - เราใช้คน 5 คนทำนาน 10 เดือน (Effort=?)
 - เรามีค่าใช้จ่ายเท่าไรเป็นอย่างน้อย (Labor Cost)
 - รวมค่าใช้จ่ายอื่นๆ ที่คำนวณง่าย เช่น ค่า License ค่าเช่าอุปกรณ์ รายจ่ายคงที่อื่น ๆ เป็นต้น
 - รวมกับ กำไรที่อยากได้

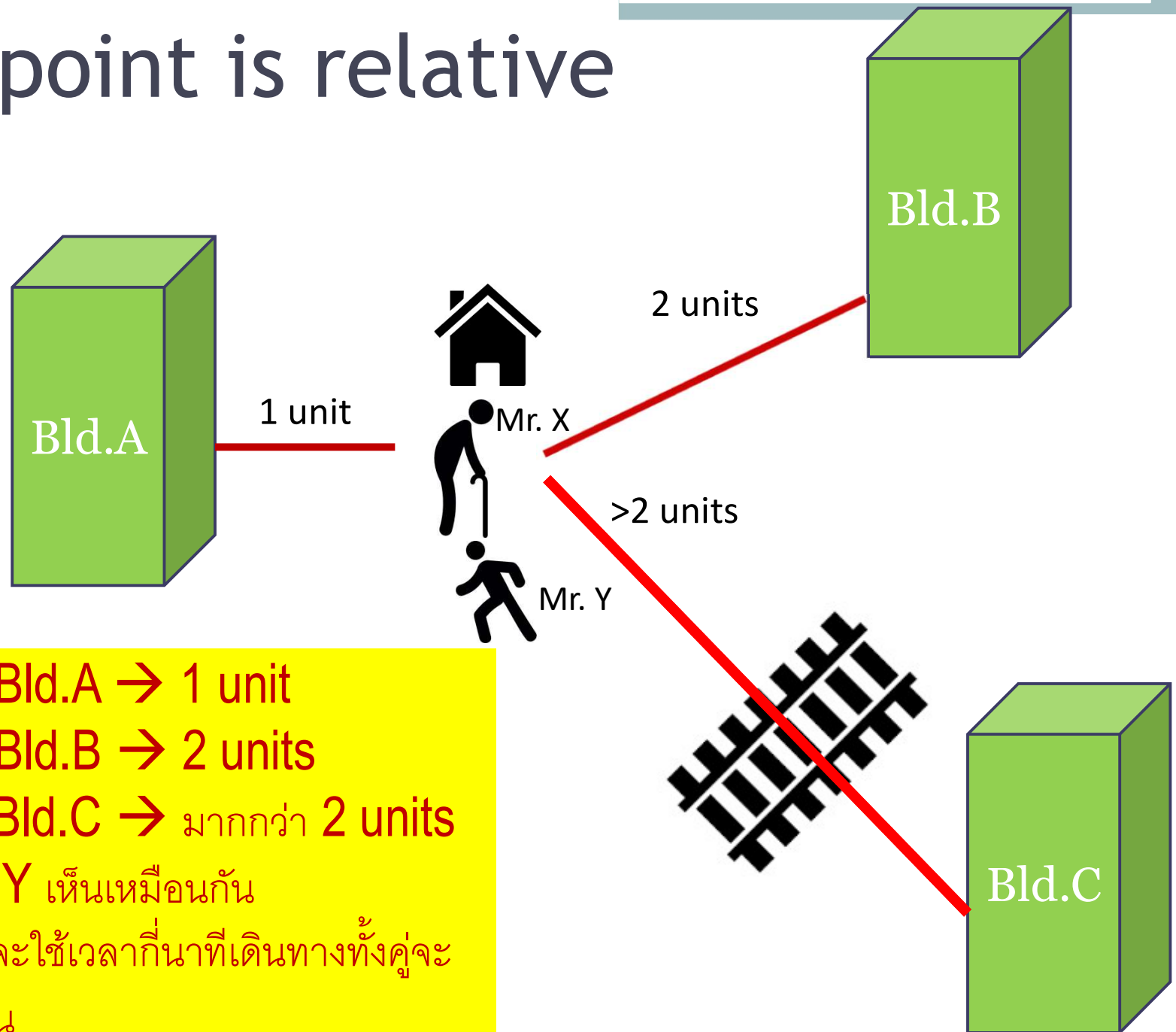
Estimate **Size** then Derive Duration

(หา **size** ก่อนแล้วค่อยไปคำนวณหา **Time** เวลาที่ใช้)



แต่ละคน **velocity** จากต่างกันได้ แทนที่เราจะประเมินเวลาที่ใช้ในการทำงาน
เราน่าจะประเมินความพยายามที่ใช้ที่เป็น **relative** มากกว่า
และเรามักจะใช้ตัววัด **size** เป็น **story point** เพราะมัน **relative** กว่า

Story point is relative



จาก Home ถึง Bld.A → 1 unit
จาก Home ถึง Bld.B → 2 units
จาก Home ถึง Bld.C → มากกว่า 2 units
Mr. X และ Mr. Y เห็นเหมือนกัน
แต่ถ้าถามใหม่ว่าจะใช้เวลากี่นาทีเดินทางทั้งคู่จะ
เห็นต่างกันแน่นอน

Story Points

- Influenced by complexity, uncertainty, risk, volume of work, etc.
- Relative values are what is important:
 - A login Screen is a 2
 - A search feature is an 8
- Basic math properties should hold (เลือกแบบคำนวณง่าย)
 - $5+4 = 9$

Estimation using Planning Poker

- A Moderator, who will not play, chairs the meeting.
- The **Product Manager** provides a short overview. The team is given an opportunity to ask questions and discuss to clarify assumptions and risks. A summary of the discussion is recorded by the Project Manager.
- Each individual lays a card face down representing their estimate. Units used vary - they can be days duration, ideal days or story points. During discussion, numbers must not be mentioned at all in relation to feature size to avoid anchoring.
- Everyone calls their cards **simultaneously** by **turning them over**.
- **People with high estimates and low estimates** are given a soap box to **offer their justification** for their estimate and then discussion continues.
- **Repeat** the estimation process until a consensus is reached. The developer who was likely to own the deliverable has a large portion of the "**consensus vote**", although the Moderator can negotiate the consensus.
- To ensure that discussion is structured; the Moderator or the Project Manager may at any point turn over the egg **timer** and when it runs out all discussion must cease and another round of poker is played. The structure in the conversation is re-introduced by the soap boxes.

Several commercially available decks use the sequence:
1, 2, 3, 5, 8, 13, 20, 40, 100, and optionally a ?
(unsure), an infinity symbol ∞ (this task cannot be
completed) and a **coffee cup** ☕ (I need a break)



Or XS, S, M, L, XL

คำแนะนำก่อนทำ **Planning poker**

- หา **story point** ที่มีค่าเป็น **2** และ **5** ก่อน (หรือ **2** และ **8** ก่อน) โดยช่วยกัน **discuss** ยังไม่ต้องเล่น **poker** ให้ **ScrumMaster** ช่วยสรุป
- จากนั้นค่อยหาที่เหลือต่อ จะทำให้มีความแม่นยำมากขึ้น และเกิด **consensus** ได้เร็วขึ้น

Exercise

- ใช้วิธี Planning Poker เพื่อทำ Consensus ในการหา Estimation ของ User story ในตารางต่อไปนี้
- เปรียบเทียบกับกลุ่มอื่นๆ ในห้อง

Output: A sample product backlog

Backlog item (EPIC#1)	Estimate (Story pt.)
As a Customer, I want to withdraw cash from an ATM, So that I don't have to wait in line at the Bank	
As a Customer, I want to deposit cash from an ATM, So that I don't have to wait in line at the Bank	
As a Customer, I want to view my account balance from an ATM, So that I don't have to wait in line at the Bank	2
As a Customer, I want to transfer cash from an ATM, So that I don't have to wait in line at the Bank	?
As a Customer, I want to pay tax from an ATM, So that I don't have to wait in line at the Bank	

กรณีที่ **Estimation** มีขนาดตัวเลขมากๆ

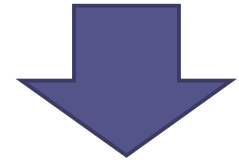
- เกิด **Product Uncertainty**
- เกิด **Technical Uncertainty**
- ให้หยุดการทำ **Estimation** ถ้าเกิด **Uncertainty** มากๆ แล้วไป **Inspection** ก่อน
- **Technical Uncertainty** อาจจะทำ **Spike** ดูก่อนได้
- **Product Uncertainty** ต้องไปปรึกษากับ **Product owner** เพื่อปรับ **User Story**

Sprint Planning



Sprint Planning

- Team selects items from the product backlog they can commit to completing
- Sprint backlog is created
 - Tasks are identified and each is estimated (1-16 hours)
 - Collaboratively, not done alone by the ScrumMaster



As a Customer, I want to withdraw cash from an ATM, So that I don't have to wait in line at the Bank



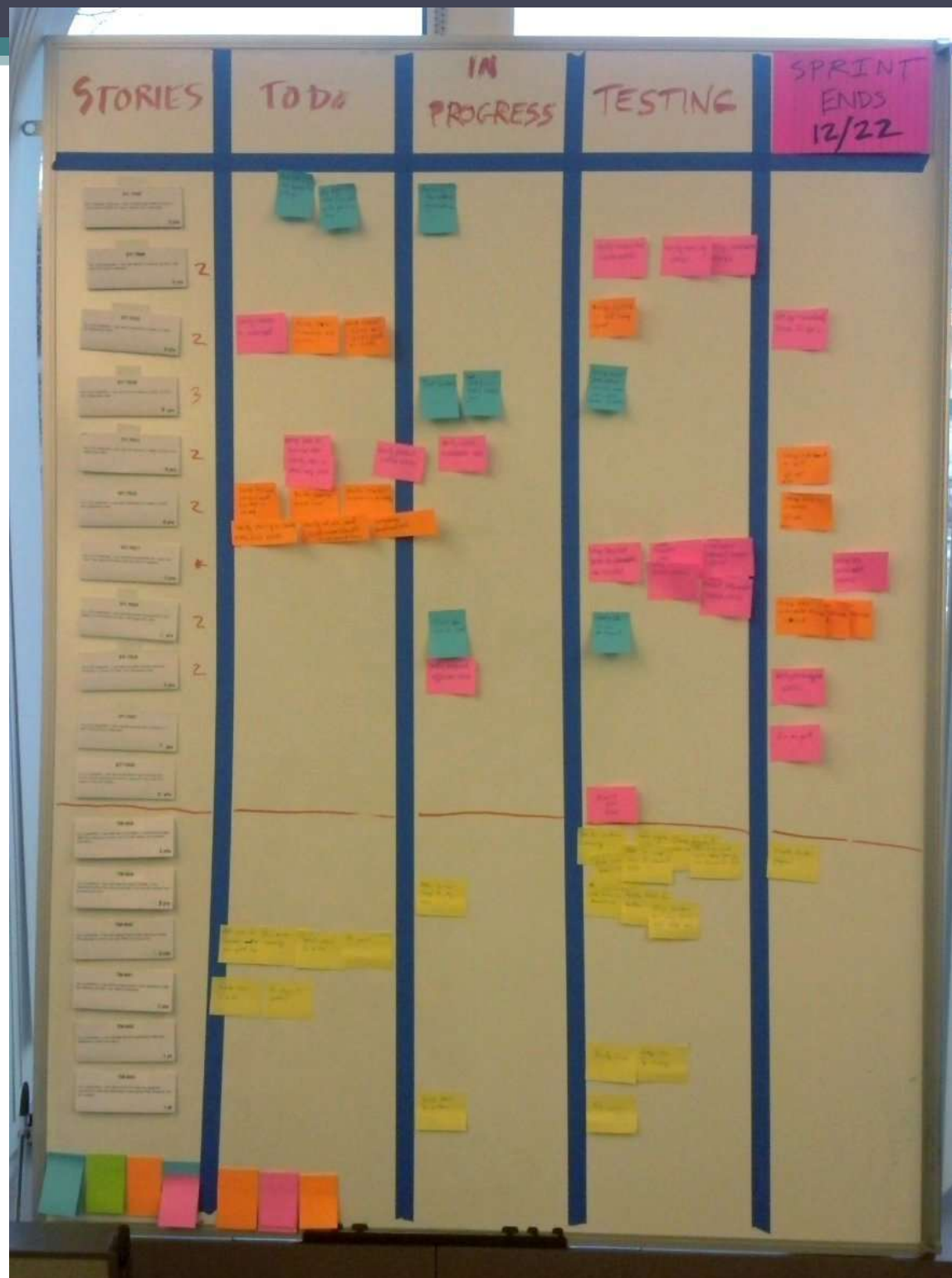
Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

During the Scrum sprint, team members are expected to update the sprint backlog as new information is available, but minimally once per day

User Story	Tasks	Day 1	Day 2	Day 3	Day 4	Day 5	...
As a member, I can read profiles of other members so that I can find someone to date.	Code the ...	8	4	8	0		
	Design the ...	16	12	10	4		
	Meet with Mary about ...	8	16	16	11		
	Design the UI	12	6	0	0		
	Automate tests ...	4	4	1	0		
	Code the other ...	8	8	8	8		
As a member, I can update my billing information.	Update security tests	6	6	4	0		
	Design a solution to ...	12	6	0	0		
	Write test plan	8	8	4	0		
	Automate tests ...	12	12	10	6		
	Code the ...	8	8	8	4		

Scrum Board

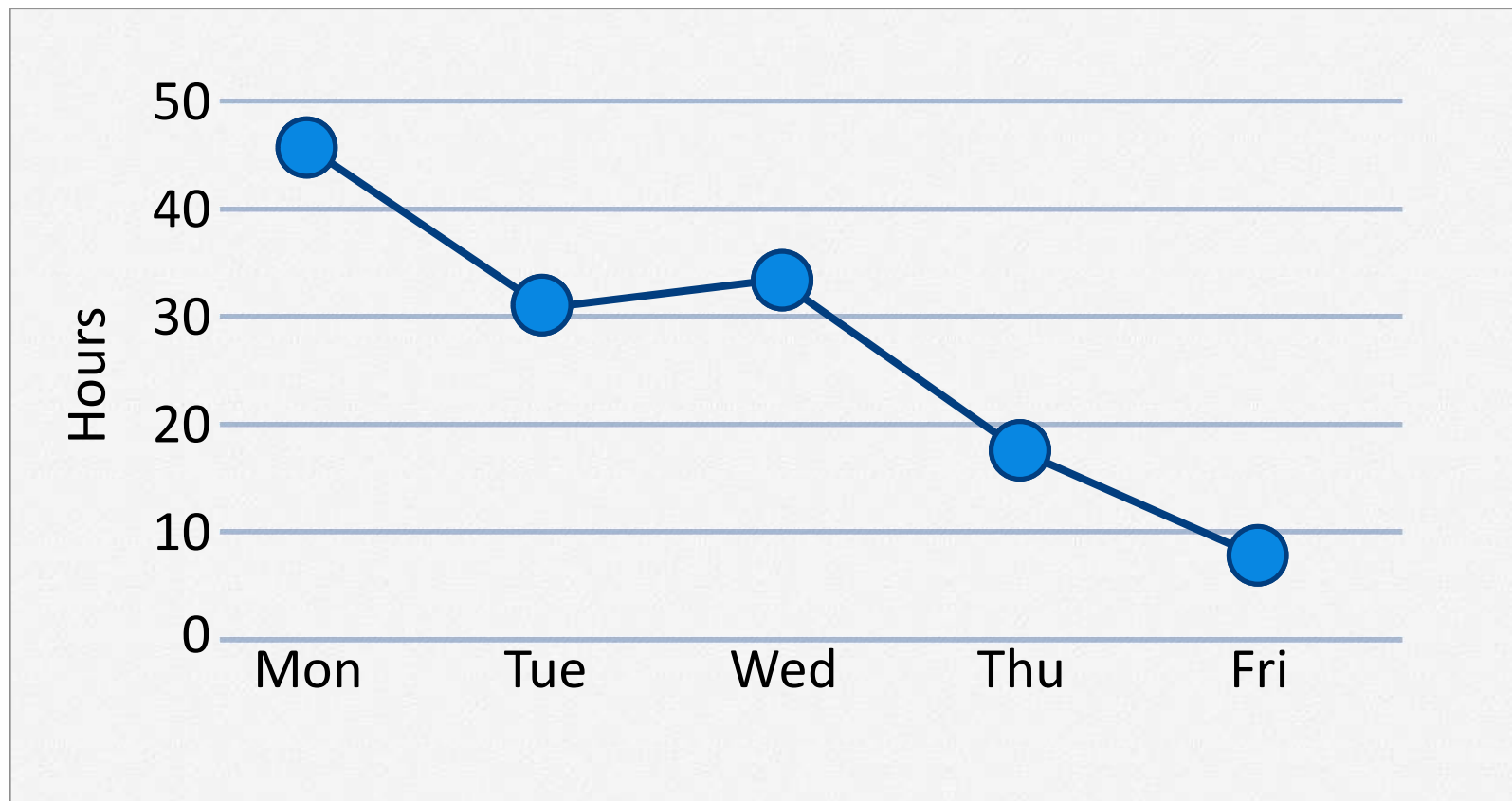
Story	To Do		In Process	To Verify	Done
As a user, I... 8 points	Code the... 9	Test the... 8	Code the... DC 4	Test the... SC 6	Code the... D
	Code the... 2	Code the... 8	Test the... SC 8		Test the... SC 8
	Test the... 8	Test the... 4			Test the... SC
As a user, I... 5 points	Code the... 8	Test the... 8	Code the... DC 8		Test the... SC
	Code the... 4	Code the... 6			Test the... SC 6



During the Sprint

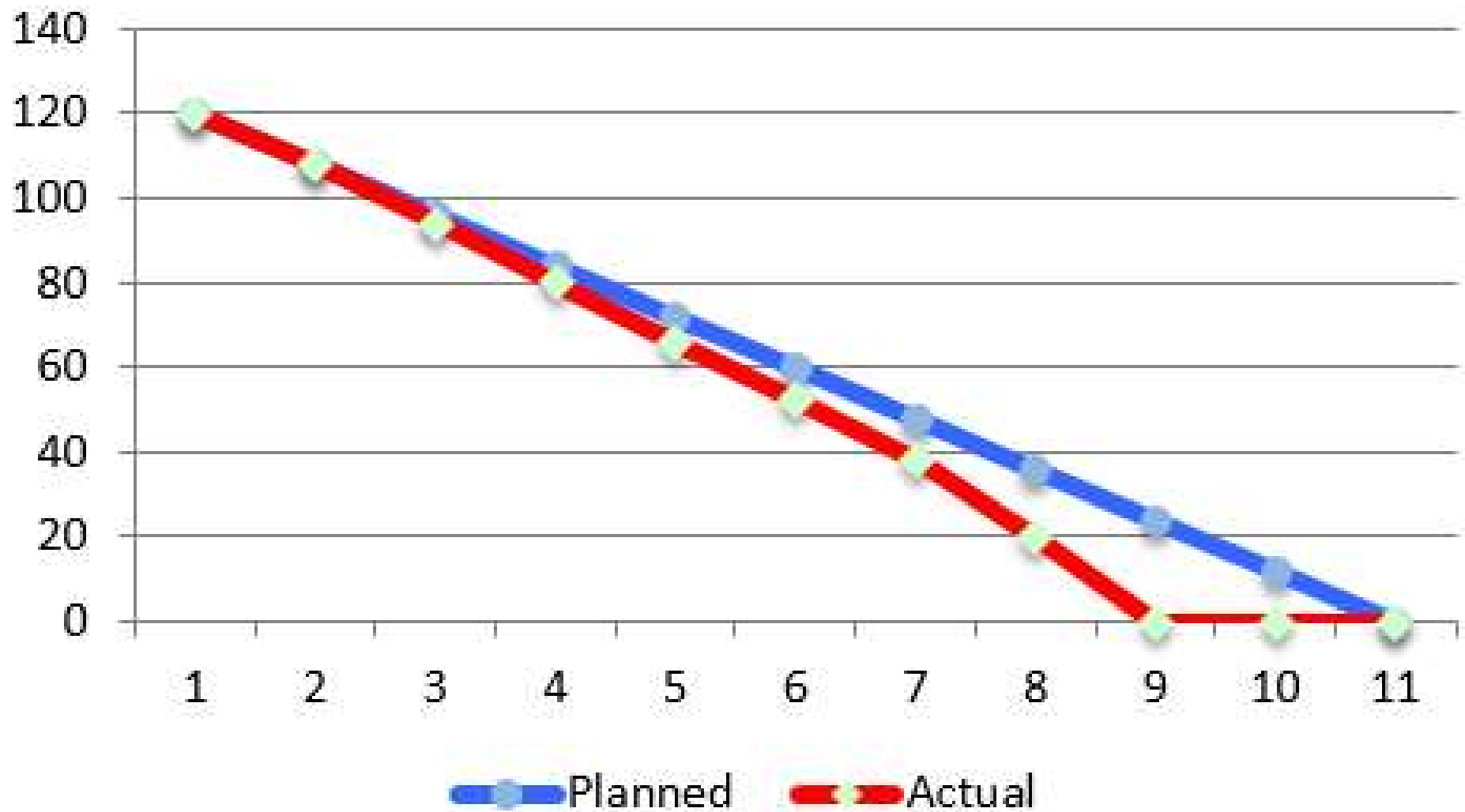
- A Task can be revised and get a new estimation

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				



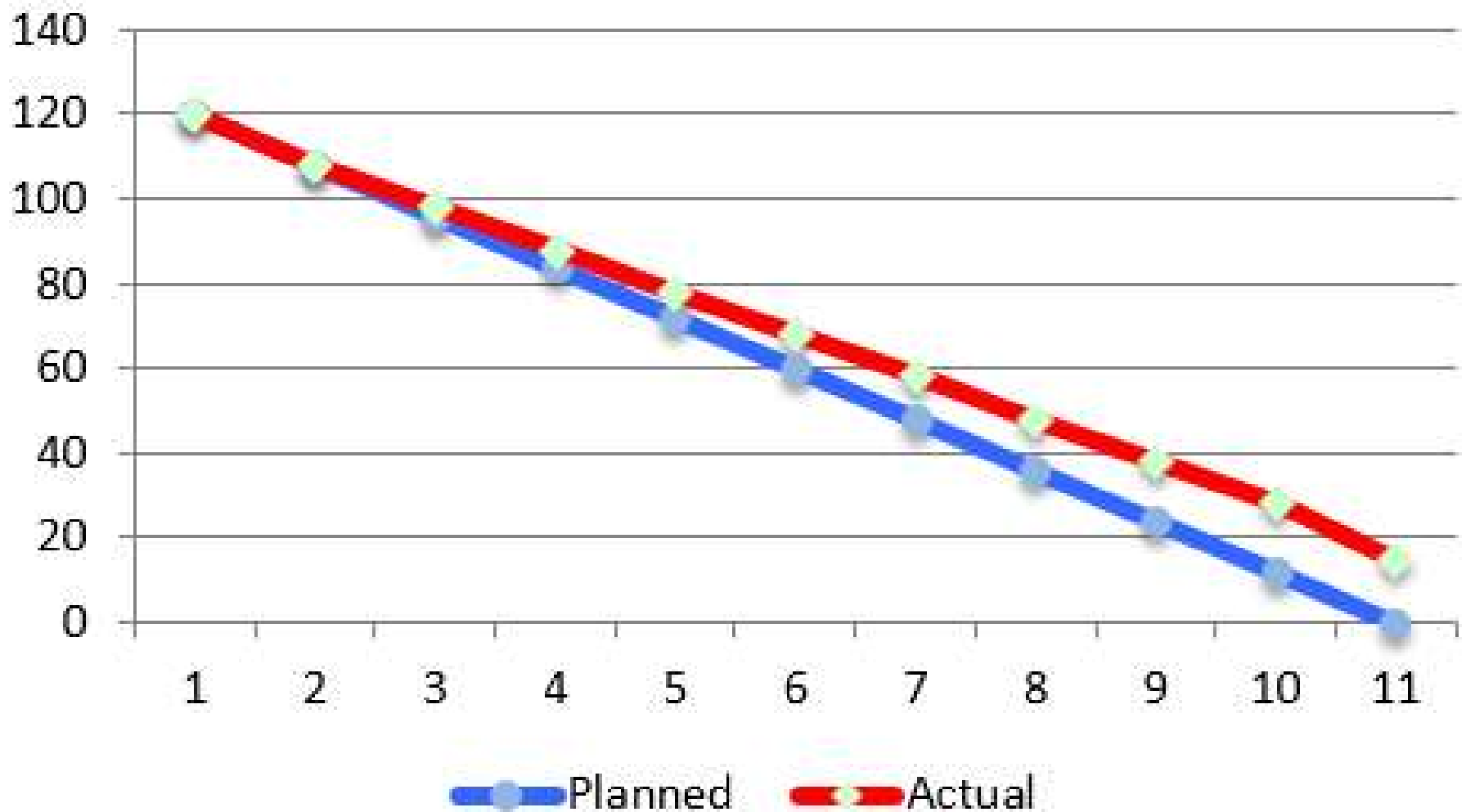
เกิดเหตุการณ์อะไร

Sprint 2 Burndown



เกิดเหตุการณ์อะไร

Sprint 2 Burndown



Daily Scrum 15 Minutes

1

What did you do yesterday?

2

What will you do today?

3

Is anything in your way?

- These are *not* status for the ScrumMaster
- They are commitments in front of peers
- Talk to the team not ScrumMaster

Exercise (อาจจะทำกิจกรรมนี้ไม่ทัน)

- เลือก **Scrum team** ออกมาหน้าห้อง
- **ScrumMaster** ควบคุม **Standup Meeting**
- **ScrumMaster** สาธิตวิธีการถาม 3 คำถาม
- **Team** ตอบคำถามโดยไม่ใช่เป็นการรายงานต่อ **ScrumMaster** แต่เป็นการพูดต่อ **team member** ทุกคน
- ใช้เวลาสั้นๆ รวมทั้งหมดไม่เกิน 15 นาทีถ้าเป็นไปได้แล้วแยกย้ายกันไปทำงานต่อ