

CS 361 – Software Engineering

Software Effort Estimation

Kamonphop Srisopha

Churee Techawut



Faculty of Science, Chiang Mai University

คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่

Agenda

- Software Effort Estimation Methods
 - Story Points Estimation
 - Algorithmic Cost Estimation

Software Cost and Effort Estimation Methods



Software Effort Estimation Methods

เป็นกระบวนการที่องค์กรหรือบริษัทใช้เพื่อคาดการณ์ ทรัพยากร (เช่น เวลา และเงิน) และ แรงกำลัง (effort) ที่จำเป็นสำหรับการพัฒนาหรือรักษาซอฟต์แวร์ เพื่อให้สามารถวางแผนงานที่มีประสิทธิภาพ จัดสรรทรัพยากรที่เหมาะสม สร้างความมั่นใจว่าโครงการสามารถสำเร็จลุล่วงตามตารางเวลาและงบประมาณที่กำหนด

Story Points Estimation

เป็นวิธีการประมาณการที่ใช้ในการพัฒนาซอฟต์แวร์แบบ Agile และ Scrum โดยอาศัยความเข้าใจร่วมกันของทีมทั้งหมดในการประเมิน โดยจะใช้งานที่ทีมได้เคยทำไปแล้วเป็นตัวอ้างอิง

Algorithmic Cost Modeling

เทคนิคที่สร้างโมเดลแบบจำลอง cost & effort จากข้อมูลโปรเจกต์ที่ผ่านมา เพื่อทำการประมาณค่าของโปรเจกต์ใหม่

Story Points Estimation

Relative Estimation vs Absolute Estimation

รถคันนี้หนักเท่าไร? (Absolute Estimation)



ตอบให้ถูกยาก

Relative Estimation vs Absolute Estimation

รถคันไหนหนักกว่ากัน? (Relative Estimation)



VS



ตอบให้ถูกง่าย
(หลายคนเห็นด้วยกับคำตอบง่ายขึ้น)

Story Points Estimation

- **Story Points คืออะไร?**

- Story Points เป็นหน่วยการวัดที่ใช้ในการประมาณการความซับซ้อน ความยากและความไม่แน่นอนของงานที่ต้องทำ (เช่น User story) (ไม่ได้หมายถึงเวลาที่จะใช้ในการทำงาน) มันบ่งบอกความสัมพันธ์ของค่าประมาณ (relative estimate)



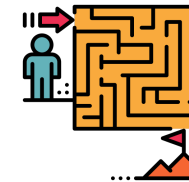
Amount of Work

พักกระดาด 10 แผ่น
เทียบกับพักกระดาด
100 แผ่น



Risk and Uncertainty

For ex. Unclear
User Story



Complexity

พักกระดาด 10 แผ่น
ให้เป็นรูปหัวใจ

ใช้สามปัจจัยข้างต้นเราตอบคำถาม “how much effort a product backlog item will require, relative to other product backlog items.”

Relative Estimations

- Story Points as Fibonacci Sequence
 - 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

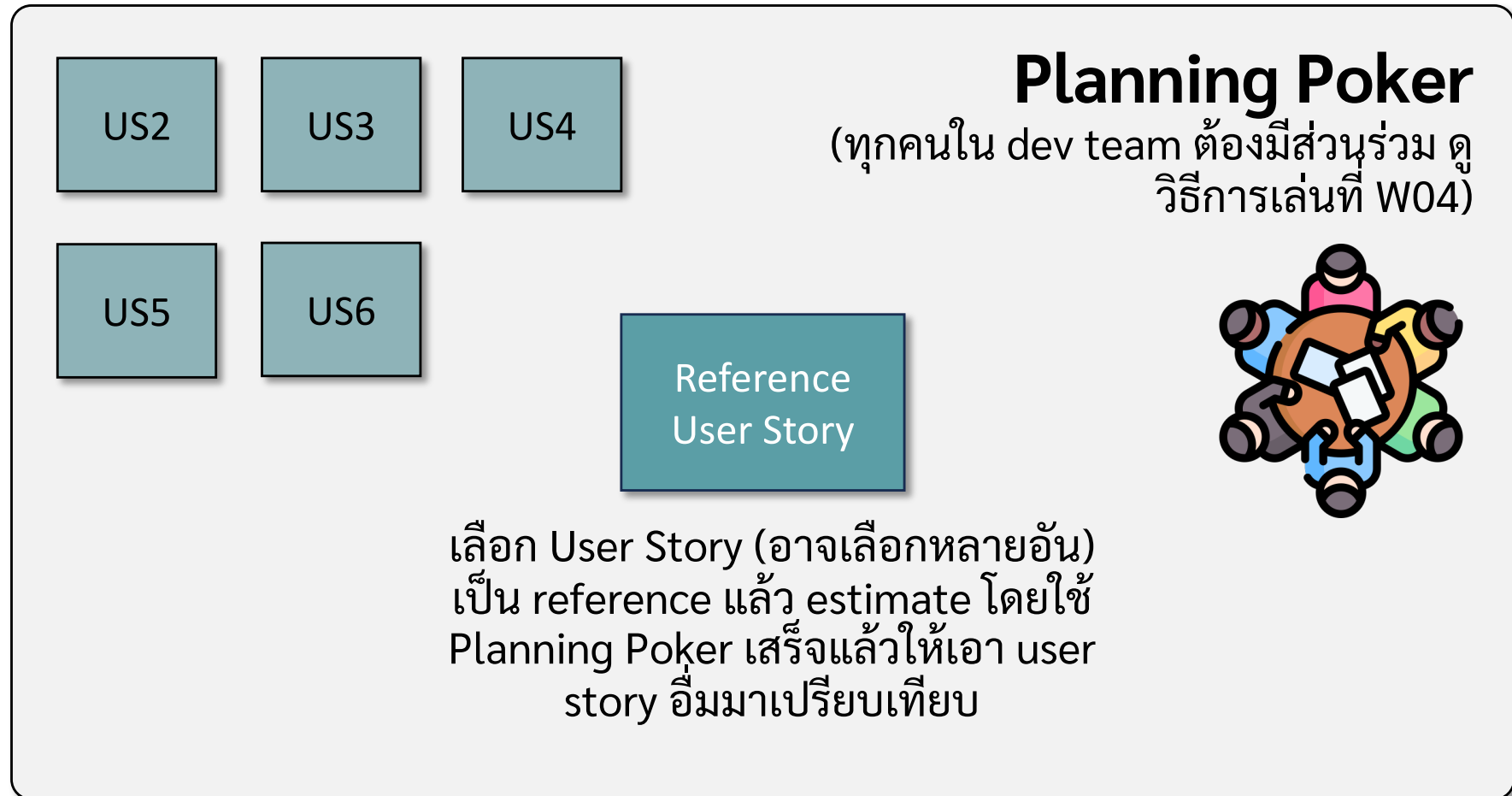


ข้อดีคือ ตัวเลขเพิ่มขึ้นอย่างเห็นได้ชัดในขั้นที่มากขึ้น แต่ % การเพิ่มนั้นของเลขจะคงที่อยู่ที่ประมาณ 60% หากใช้ sequence ที่เป็น linear scale ก็จะไม่สามารถบอกได้ว่า User Story นี้น่า 44, 46, หรือ 48 (เพราะแตกต่างกันน้อย) แต่ถ้าเป็น 34, 55, 89 จะสามารถทำให้คนในที่มึนมีความเห็นที่ตรงกันได้ง่ายว่า และ สำหรับงานขนาดเล็ก ลำดับ Fibonacci ก็เหมาะสมเพราะงานขนาดเล็กโดยทั่วไปจะจัดการได้ง่ายและง่ายต่อการประมาณที่แม่นยำกว่าได้ (2 vs 3 vs 5)

- Story Points as T-Shirt Sizing



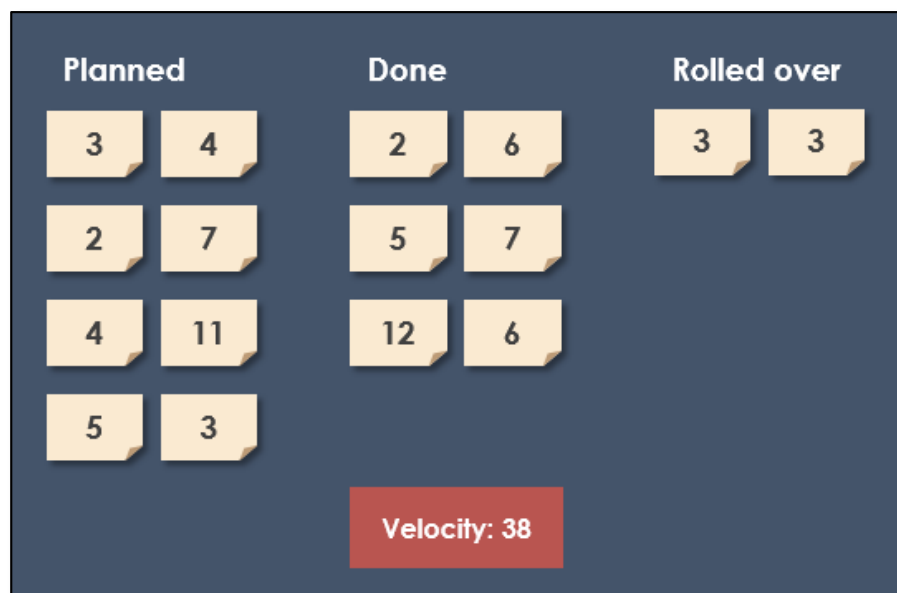
Product Backlog Prioritization



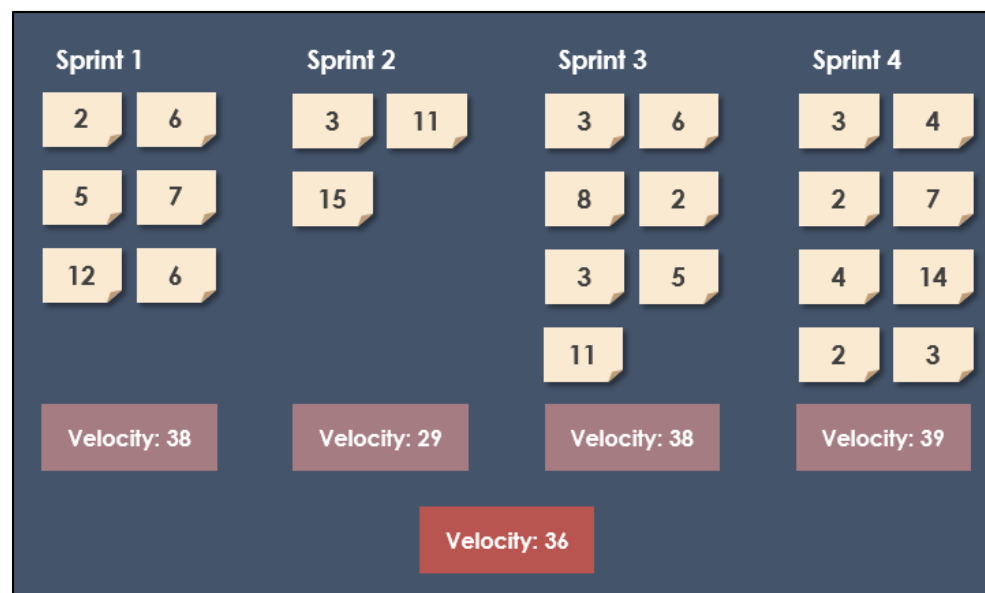
TIPS: ในการจัดลำดับความสำคัญของ User story ใน product backlog เราอาจจะทำ story point มาใช้เลยก็ได้หรือรวมกับ criteria อื่นๆ เช่น Business Value (แล้ว prioritize โดยใช้ Multiple-Criteria Decision Analysis หรือวิธีอื่นๆ)

Story Points to Velocity

Velocity เป็น Metric หนึ่งในกระบวนการ scrum บ่งบอกปริมาณงานที่ทีมสามารถทำให้สำเร็จได้ภายในรอบการทำงานหนึ่ง (Sprint) วัดจากค่ารวม "Story Points". เป็นวิธีที่ทีมใช้เพื่อคาดการณ์ความสามารถในการทำงานของ ทีมในอนาคต จากปริมาณงานที่ทีมได้ทำสำเร็จในรอบ sprint ก่อนหน้า



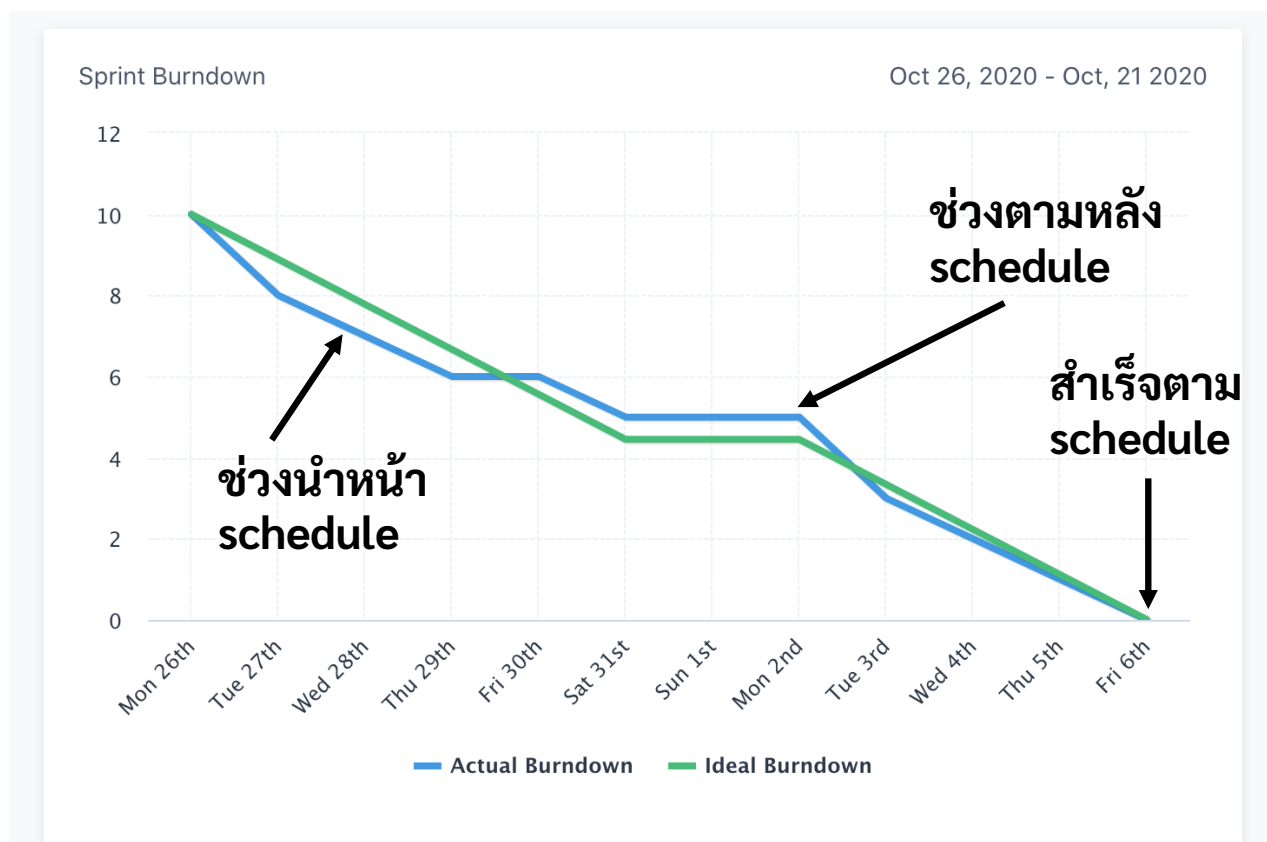
ในหนึ่ง Sprint จะนับ Velocity จาก story points ที่ทีมสามารถทำได้สำเร็จ (“**DONE**”)



ใช้ค่า average ของค่า velocity ของ sprint ก่อนหน้ามาประมาณ Story points รวมที่ทีม คาดว่าน่าจะได้ใน Sprint ต่อไป

Story Points to Burn Down Chart

Burndown Chart เป็นเครื่องมือที่สามารถช่วยให้ทีมเห็นถึงความคืบหน้าและประเมินสถานะของโปรเจกต์ในเวลาที่เหมาะสม และสามารถประเมินได้ว่าจะสามารถเสร็จงานตามกำหนดหรือไม่ หรือว่าอาจจะต้องปรับแผนใหม่

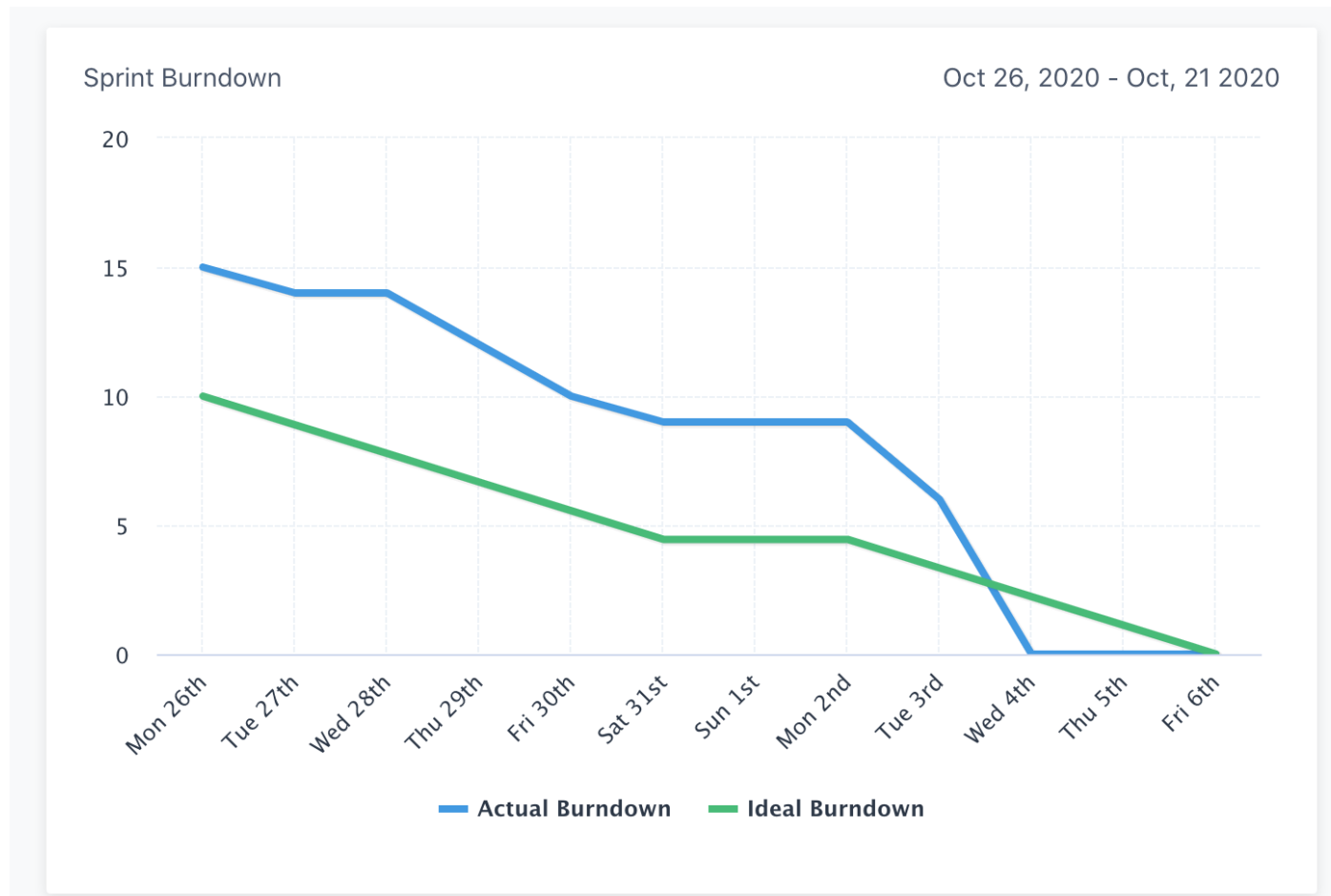


แกน x แสดงช่วงเวลา
แกน y แสดงจำนวนงานที่ยังเหลืออยู่ (อาจเป็น ชั่วโมง, task, story points, เป็นต้น)

เส้นสีเขียวคือเส้น ideal งานที่เหลืออยู่แบบ constant

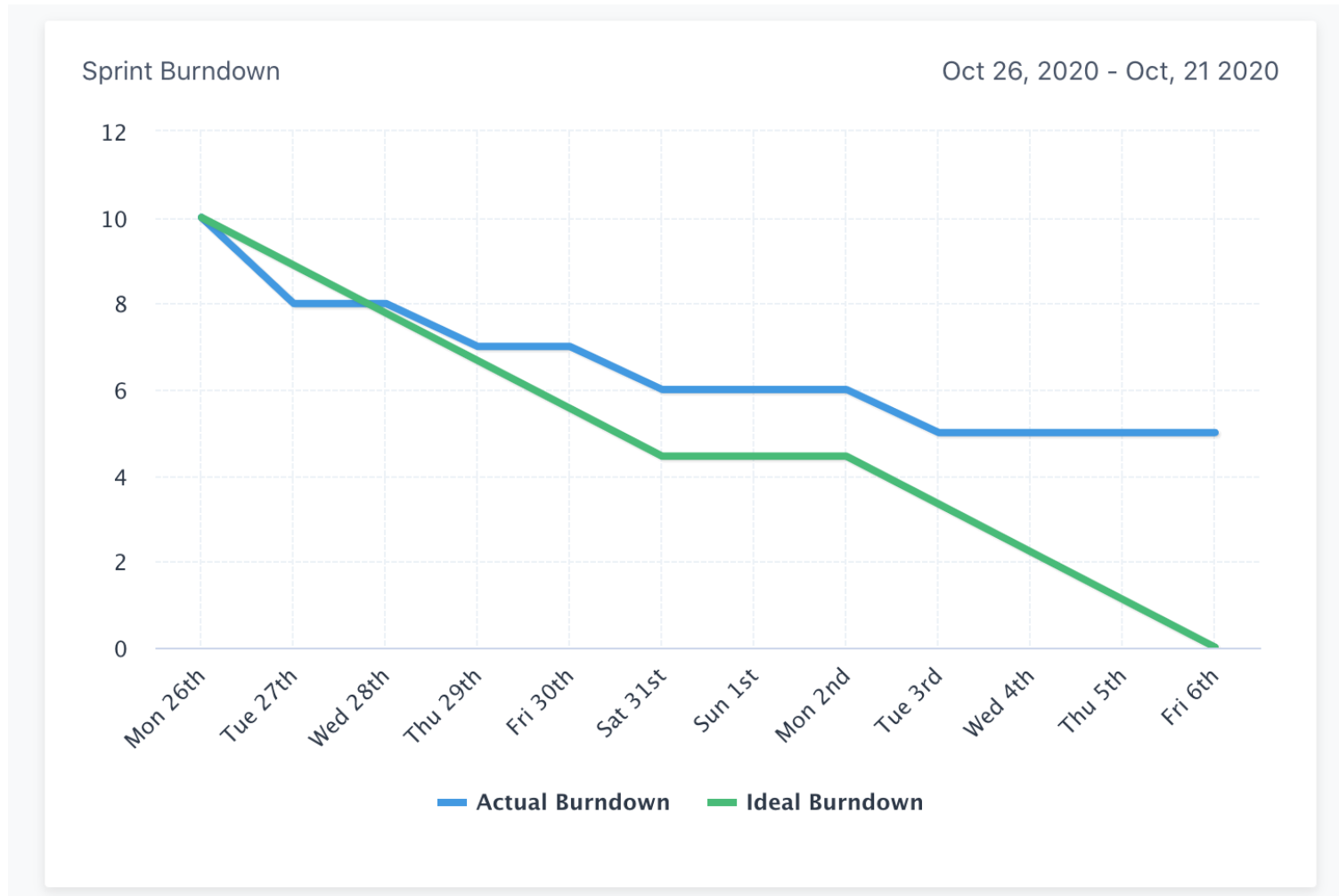
เส้นสีฟ้าคือเส้น actual งานที่เหลืออยู่จริงๆในแต่ละวัน

Example: Burn Down Chart



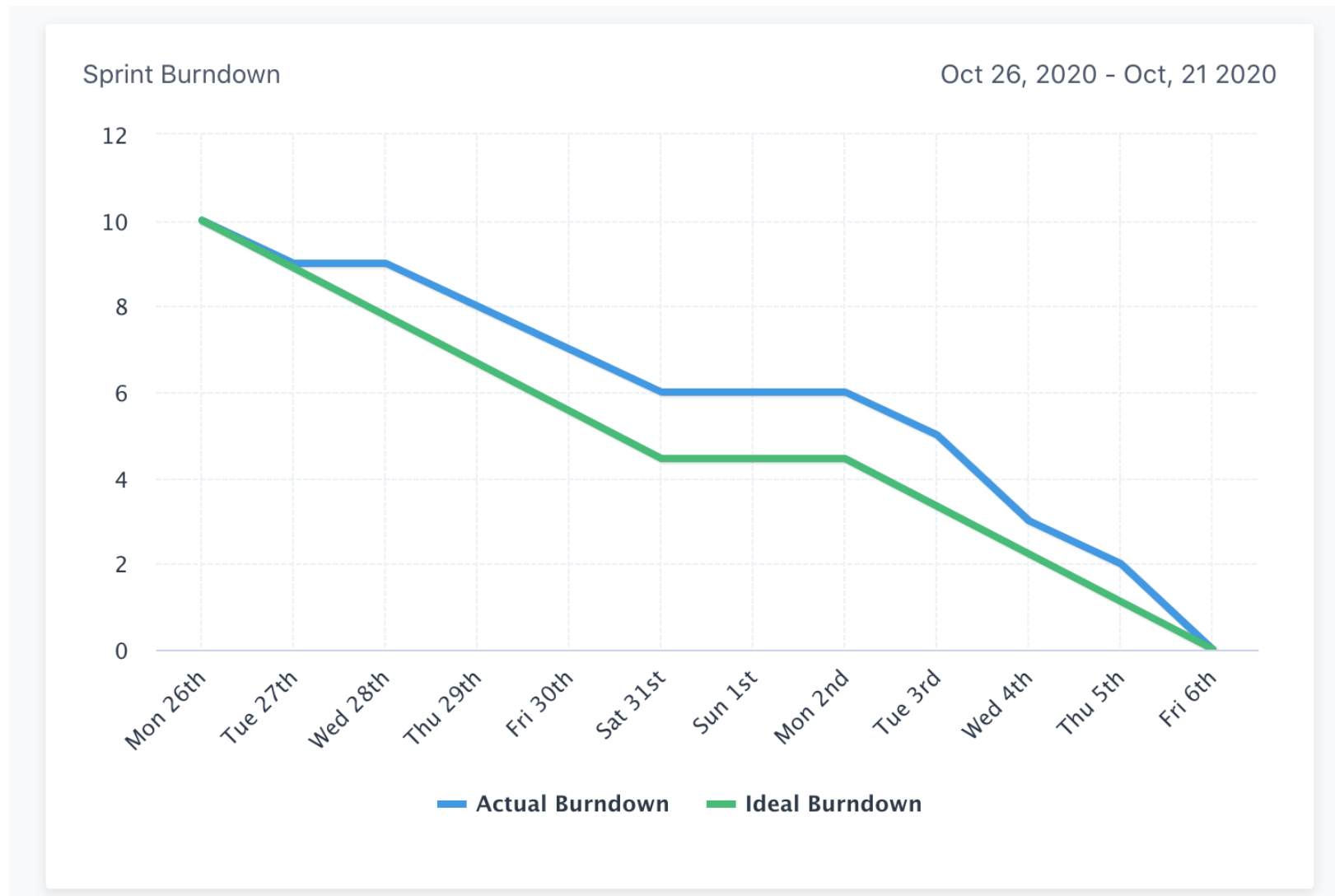
จากกราฟใน Sprint นี้ทีมทำงานล่าช้ากว่าที่วางไว้ในแต่ละวัน
(งานแต่ละวันเหลืออยู่เยอะ) แต่ก็สามารถเสร็จงานก่อนกำหนด
(อาจเกิดจาก undercommitting)

Example: Burn Down Chart



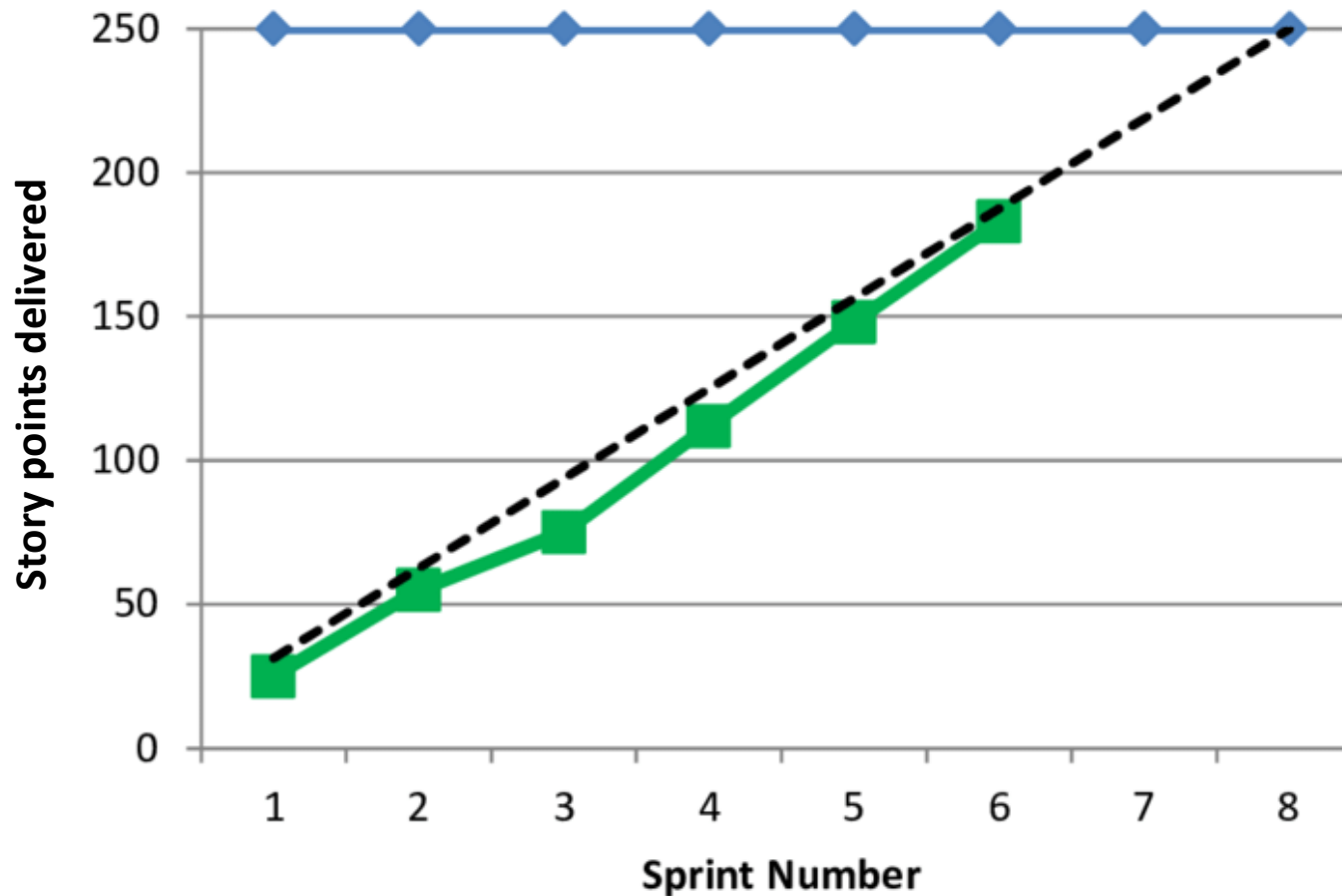
เมื่อเห็นกราฟนี้ นักศึกษาสามารถบอกอะไรเกี่ยวกับทีมนี้ได้ ?

Example: Burn Down Chart



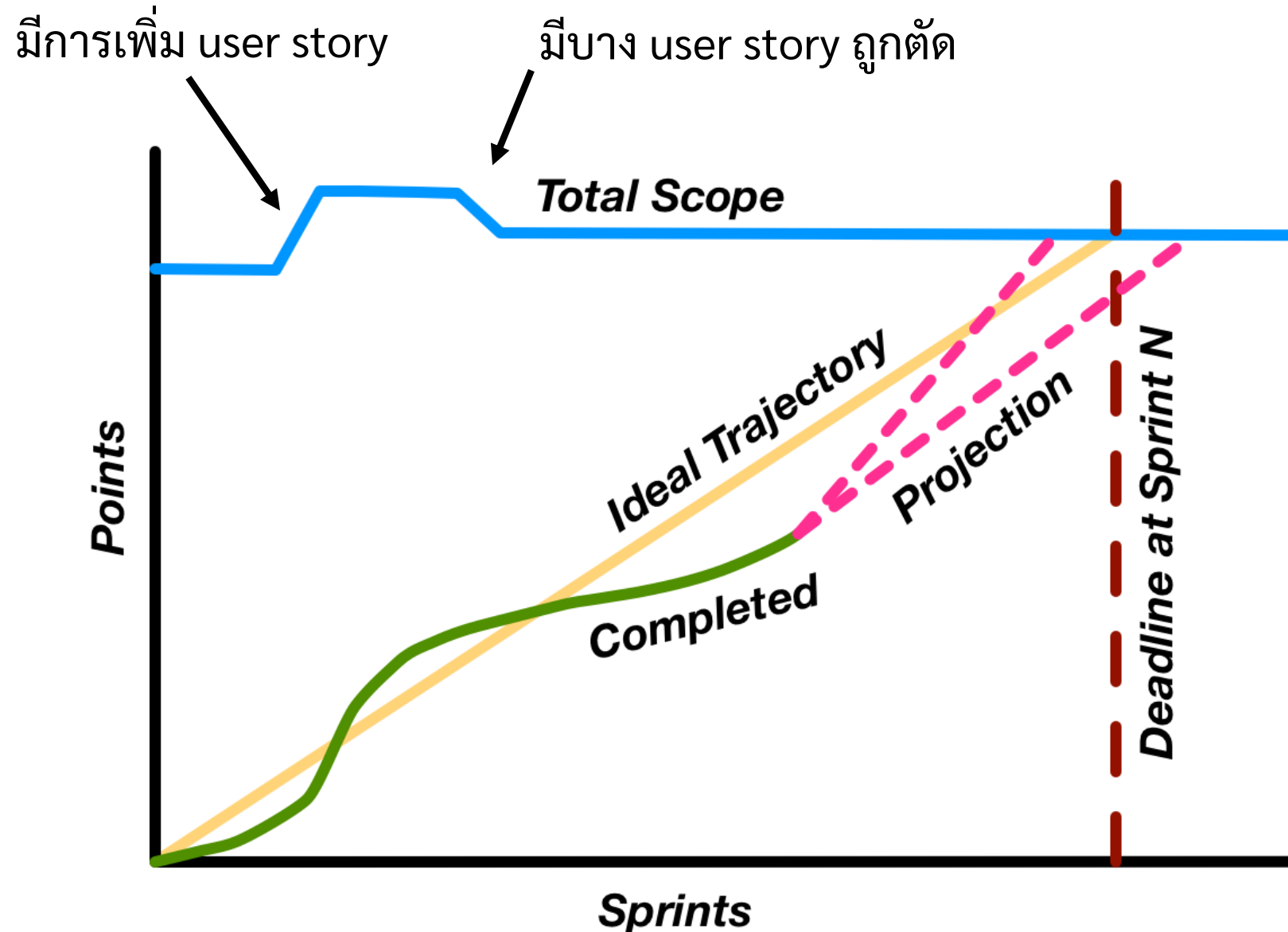
เมื่อเห็นกราฟนี้ นักศึกษาสามารถบอกอะไรเกี่ยวกับทีมนี้ได้ ?

Story Points to Burn Up Chart



เส้น **สีฟ้า** จำนวน Story Points ทั้งหมด
เส้น **สีเขียว** คือจำนวน Story Points ที่ส่งมอบรวม
เส้น **ดำประ** คือเส้น ideal

Story Points to Burn Up Chart



Advantages and Limitations of Story Points

- เห็นความสัมพันธ์ระหว่าง User Story ได้ชัด (เช่น A ใช้ effort มากกว่า B สองเท่า)
- เป็นกุศโลบายที่ส่งเสริมการสนทนา ซึ่งนำไปสู่ความเข้าใจ user story ที่ดีขึ้นในทีมและเปิดเผยปัญหาที่ซ่อนอยู่ได้ (เช่น ถ้า effort ในการทำ story นี้เยอะก็ควรจะแตก story ออกมาเป็นย่อยๆ)
- นำมาเป็นตัวช่วยในการประมาณได้ (Story Point ที่ทีมสามารถทำได้ใน 1 Sprint – เรียกว่า Velocity) ใน sprint ถัดไปก็จะเห็นได้ว่ามี user story ไหนที่ทีมทำได้

- ขึ้นอยู่กับความเข้าใจของคนในทีม
- ไม่ได้ประเมินเวลาที่ใช้ได้โดยตรง
- ไม่สามารถเปรียบเทียบระหว่างทีมได้
- ไม่ง่ายที่จะหา user story ที่จะให้มาเป็น reference user story
- อาจใช้เวลาค่อนข้างสูงในการทำ planning poker หากมี user stories ใน backlog เยอะ