



**FPT POLYTECHNIC**

QUẢN LÝ DỰ ÁN VỚI AGILE

**Bài 5: Lập kế hoạch và thực  
thi dự án**

---

[www.poly.edu.vn](http://www.poly.edu.vn)

---

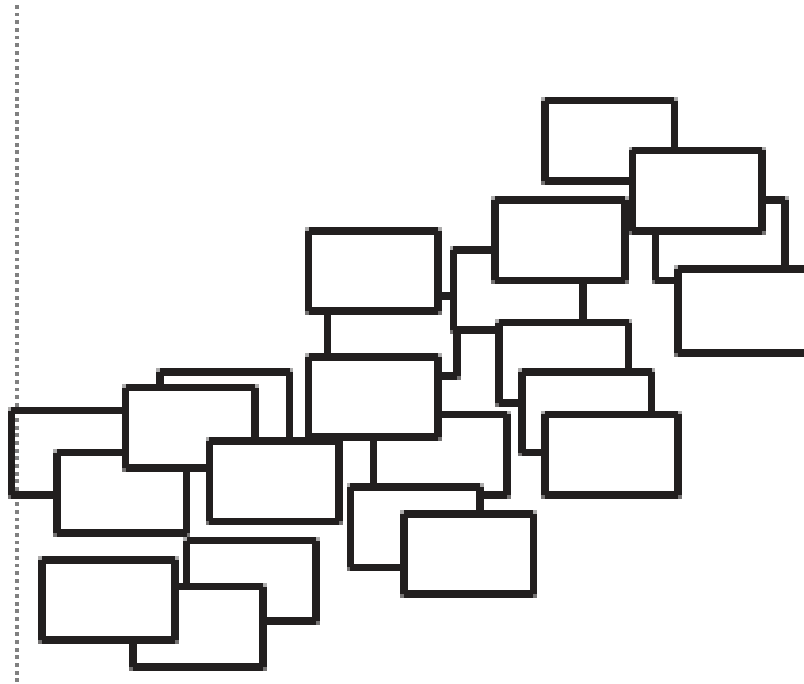
## Nội dung bài học

- Từ tầm nhìn kiến trúc đến lập kế hoạch phát hành và lập kế hoạch sprint
- Từ phát triển phần mềm tăng trưởng đến phát triển phần mềm song song
- Biểu đồ Burndown
- Tốc độ nhóm



# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

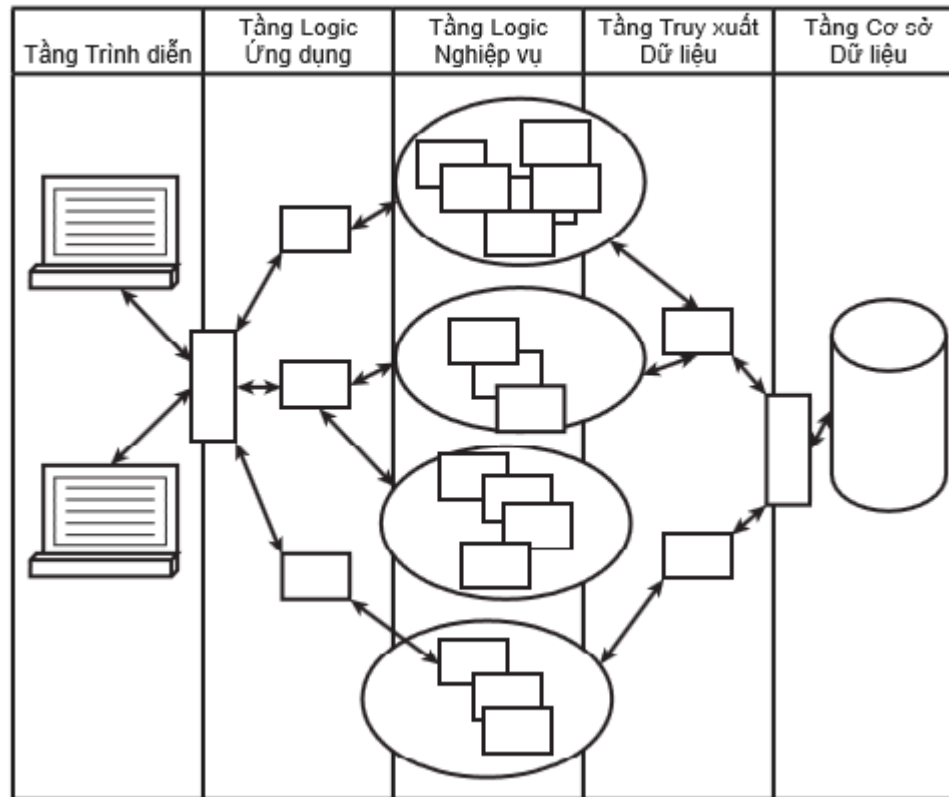
- Thay vì thụ động để cho Product owner quyết định những gì mà nhóm sẽ làm trong một Sprint, nhóm sẽ có thêm nhiều lợi ích từ việc cộng tác với Product Owner



Thanh tra tập các user story cho một hệ thống thư viện trung tâm

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

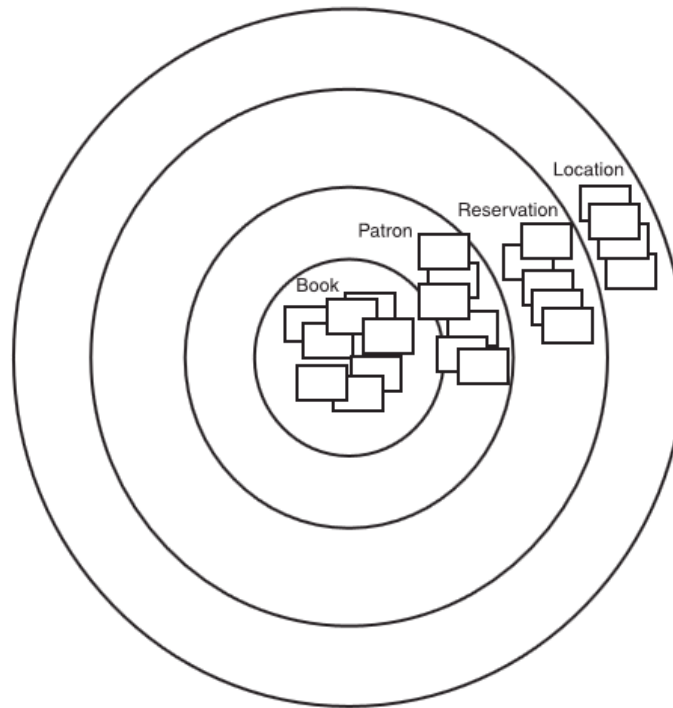
Nhóm có thể lướt qua tập user story và xác định được một kiến trúc ở mức cao như sau:



Xác định ý đồ kiến trúc cho hệ thống thư viện trung tâm

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

- Sử dụng cách tiếp cận các thành phần dữ liệu chung, nhóm có thể phân chia các user story thành các vòng khác nhau

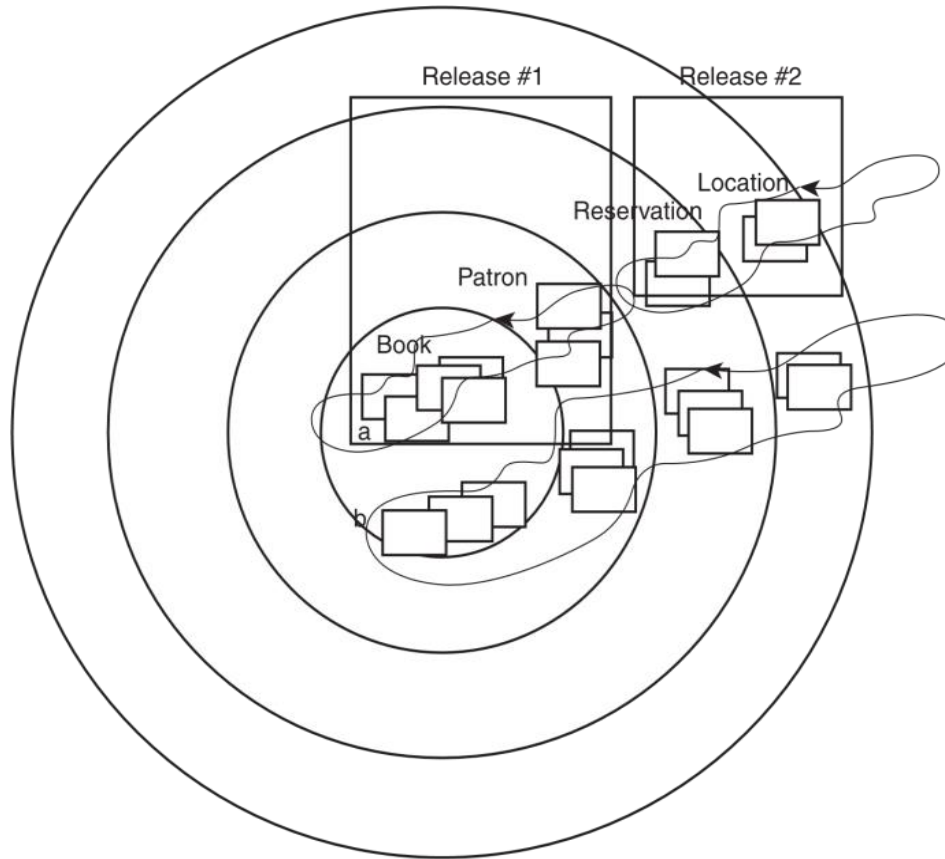


Tiếp tục phân chia các User Story theo các thành phần dữ liệu chung

# **Từ Tâm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint**

- Lát cắt chiều ngang có nghĩa là nhóm có thể hình dung được việc phát triển sản phẩm phần mềm bằng cách tạo nền tảng cho những thành phần dữ liệu và thực thể quan trọng trên toàn bộ các vòng, bắt đầu từ phần lõi ở giữa cho đến vòng ngoài cùng
- Trong trường hợp của hệ thống thư viện trung tâm, việc này có thể mang ý nghĩa là đặt nền tảng cho các thực thể Book và Patron đầu tiên trong bản phát hành số 1, còn Reservation và Location thì trong bản phát hành số 2

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint



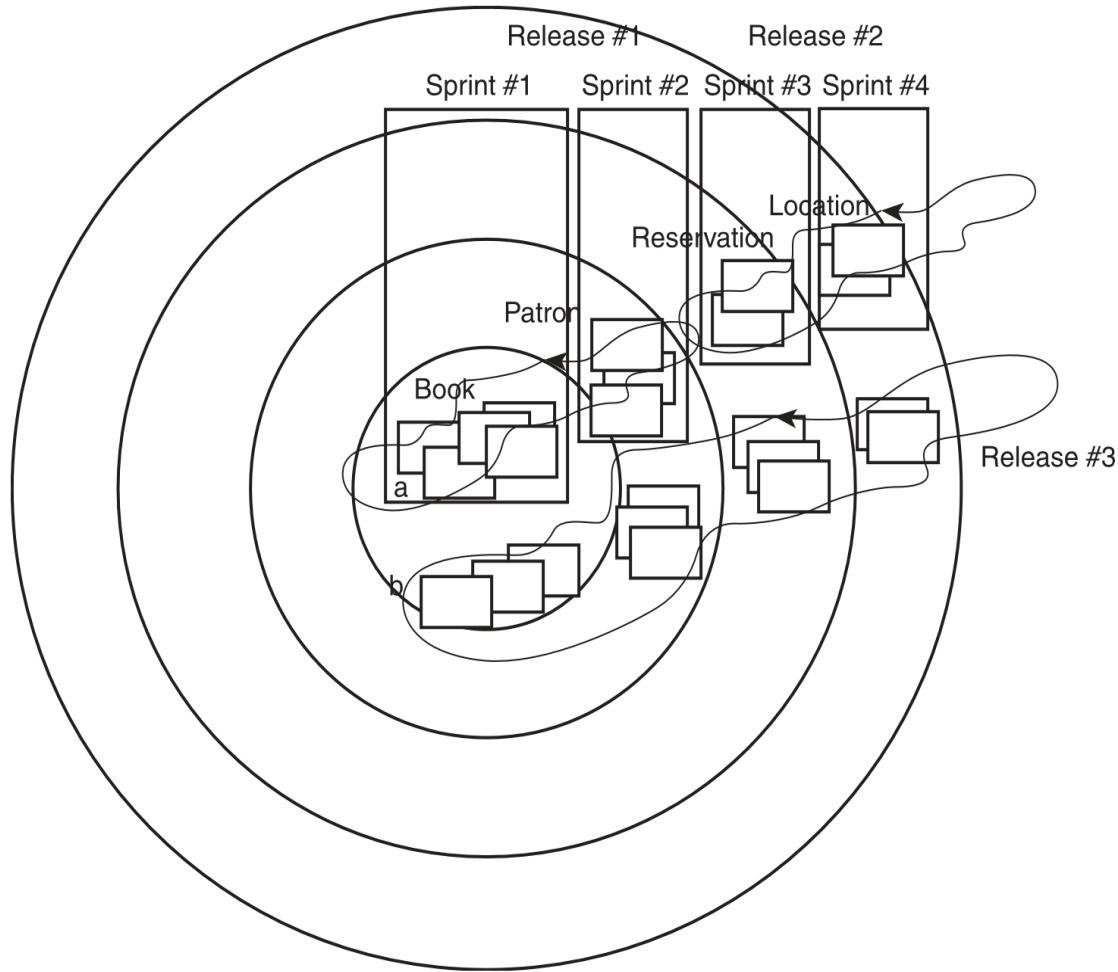
Tổ chức bản phát hành theo lát cắt ngang

# Từ Tâm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

- Tiếp theo, tùy theo tính chất của dữ liệu chung (và có thể cả tốc độ của nhóm) mà bản phát hành số 1 có thể được chia thành hai Sprint.
  - Sprint 1 sẽ tập trung vào phần đầu tiên của khái niệm Book
  - Sprint 2 thì tập trung vào phần đầu tiên của khái niệm Patron
- Có thể áp dụng cách tiếp cận đó cho bản phát hành số 2. Nó cũng được chia thành hai Sprint
  - Sprint 3 tập trung vào khái niệm Reservation
  - Sprint 4 nhấn mạnh vào khái niệm Location



# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

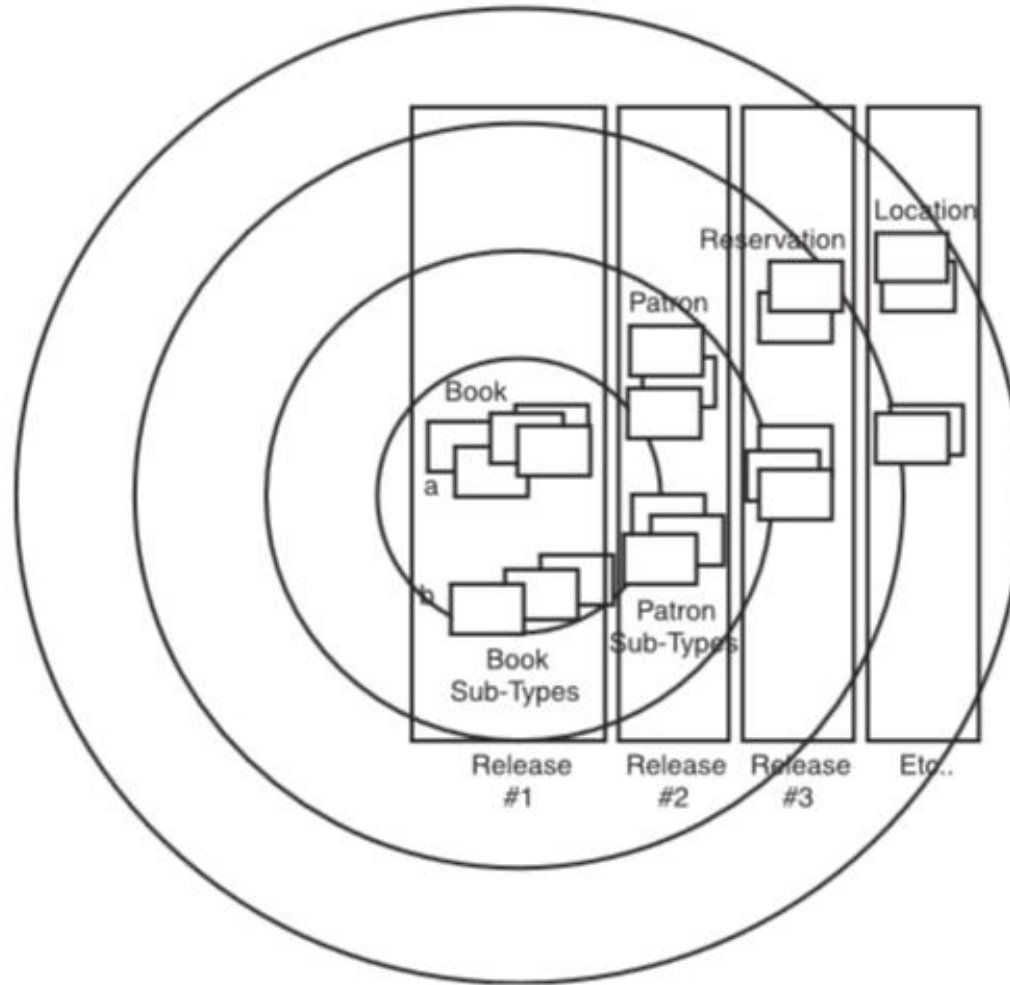


Tổ chức bản phát hành theo lát cắt dọc

# **Từ Tâm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint**

- Lát cắt chiều dọc có nghĩa là nhóm có thể phát triển sản phẩm phần mềm bằng cách tạo nền tảng cho tất cả các thành phần dữ liệu và thực thể, từng vòng một
- Trước tiên nhóm có thể đặt nền tảng cho tất cả mọi thứ liên quan đến khái niệm Book ngay ở trong bản phát hành số 1
- Sau đó, tất cả các chi tiết của khái niệm Patron có thể ở trong bản phát hành số 2, và cứ tiếp tục như vậy

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

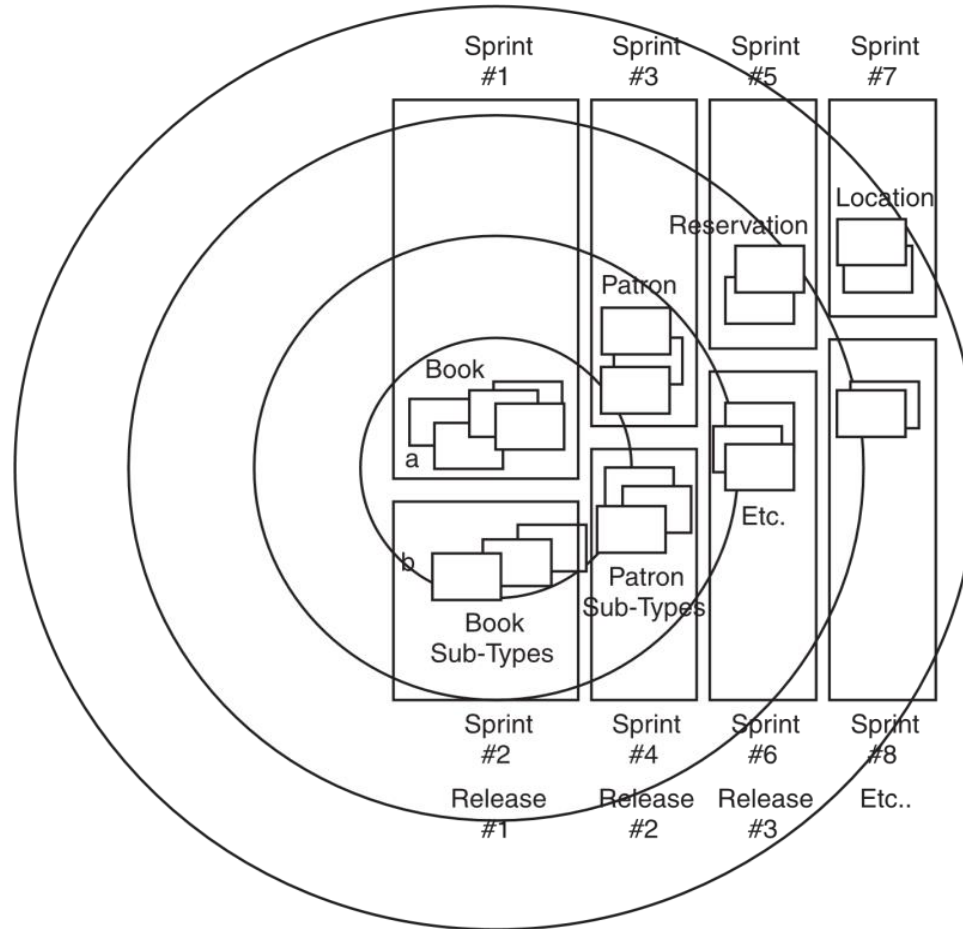


Chia bản phát hành thành các Sprint

# **Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint**

- Có thể áp dụng cách làm này cho bản phát hành số 2, như được thể hiện trong hình trước.
- Có thể được chia làm hai Sprint sau:
  - Sprint 3 tập trung vào các phần cơ bản của khái niệm Patron
  - Sprint 4 tập trung các phần mở rộng của nó vào những khía cạnh của các kiểu con của khái niệm Patron,...

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint



Chia bản phát hành thành các Sprint

# **Từ Tâm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint**

- Sau một thời gian, bạn sẽ nhận ra rằng mình không theo dõi được tất cả các mục đích của bản phát hành và của Sprint, có thể bạn muốn sắp xếp chúng, kèm theo các mô tả ở mức cao tương ứng của các Story bằng cách sử dụng một ma trận như trong hình sau
- Ngoài việc mang lại lợi ích cho nhóm Scrum trong việc lập kế hoạch phát hành và Sprint thì lược đồ sau còn có thể dễ dàng trở thành một bản Sprint Backlog mới ,nó hỗ trợ việc phân công công việc và theo dõi tiến độ của nhóm

# Từ Tầm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

Bản phát hành	Sprint	Story	Mục tiêu
1. Bản phát hành số 1	1. Sprint 1 (Book)	1. Story 1 2. Story 2	.....  .....
2. Bản phát hành số 2	2. Sprint 2 (Patron)	3. Story 3 4. Story 4	.....

Theo dõi đối sánh các mục tiêu Sprint và mục tiêu của bản phát hành (triển khai theo lát cắt ngang)

# Từ Tâm nhìn Kiến trúc đến Lập kế hoạch Phát hành và Lập kế hoạch Sprint

				Lượng công việc còn lại theo giờ					
Backlog	Nhiệm vụ	Thành viên	Ước tính (Giờ)	Ngày 1	Ngày 2	Ngày 3	Ngày 4	Ngày 5	Ngày 6
Bản phát hành số 1	Mục tiêu:								
1. Sprint 1	Mục tiêu:								
Story 1	.Thiết kế/Viết mã Giao diện Người dùng (UI)	John	4	2	2	...	...	...	...
	.Thiết kế/Viết mã Tầng Ứng dụng	Frank	2	2	1				
	.Thiết kế và Viết mã Tầng Nghiệp vụ	Andrew	5	2	2				
	.Thiết kế và Viết mã Tầng Truy xuất Dữ liệu	Laura	6	5	6				
	.Thiết kế và Viết mã Tầng Cơ sở Dữ liệu	Lola	4	3	3				
Story n	.Thiết kế/Viết mã Giao diện Người dùng (UI)	John	7	5	...	...	...	...	...
	.Thiết kế/Viết mã Tầng Ứng dụng	Andrew	5	5	...	...	...	...	...
	.v.v..	v.v...							
2. Sprint 2	Mục tiêu:								
Story 1	.Thiết kế/Viết mã Giao diện Người dùng (UI)	John	3	2	1	...	...	...	...
		v.v...							
		Tổng cộng							

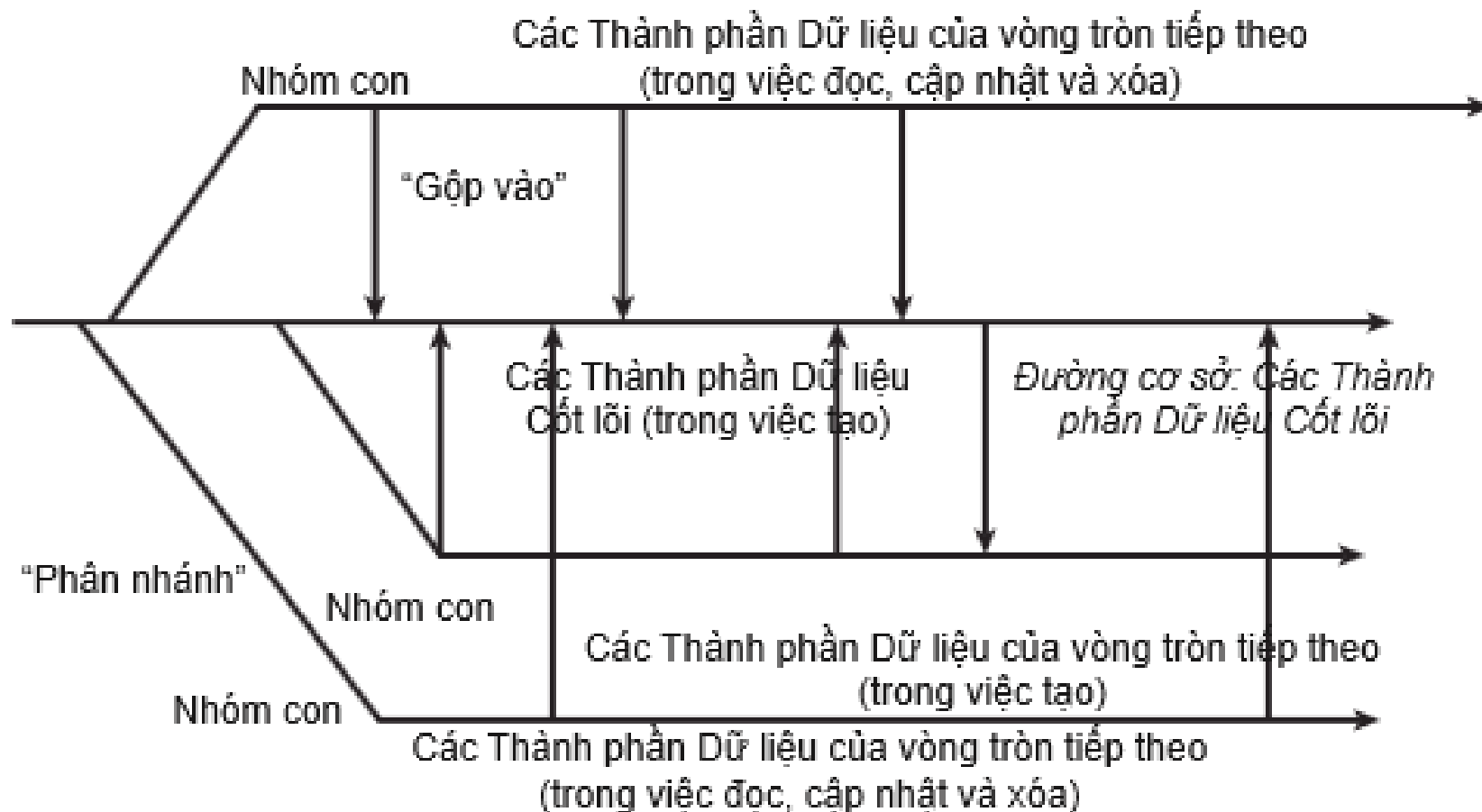
Một Sprint Backlog được tổ chức theo Sprint và bản phát hành



# **Từ Phát triển Phần mềm Tăng trưởng đến Phát triển Phần mềm Song song**

- Ngoài việc đem lại sự hỗ trợ cho nhóm Scrum về mặt lập kế hoạch phát hành và lập kế hoạch Sprint thì tầm nhìn kiến trúc còn mang lại lợi ích cho một khía cạnh khác đó là: phát triển phần mềm song song (thậm chí với một nhóm Scrum 7-thành viên)
- Thông qua việc tổ chức công việc xung quanh các thành phần dữ liệu chung và khái niệm CRUD, bạn có khả năng tăng tốc độ phát triển bằng cách cho các nhóm con nhỏ hơn làm việc song song trên các user story cụ thể và khép kín
- Việc này, đi kèm với một cơ chế tích hợp liên tục hiệu quả (sử dụng công cụ như Git), sẽ cho phép nhóm Scrum tăng tốc độ một cách đáng kể

# Từ Phát triển Phần mềm Tăng trưởng đến Phát triển Phần mềm Song song

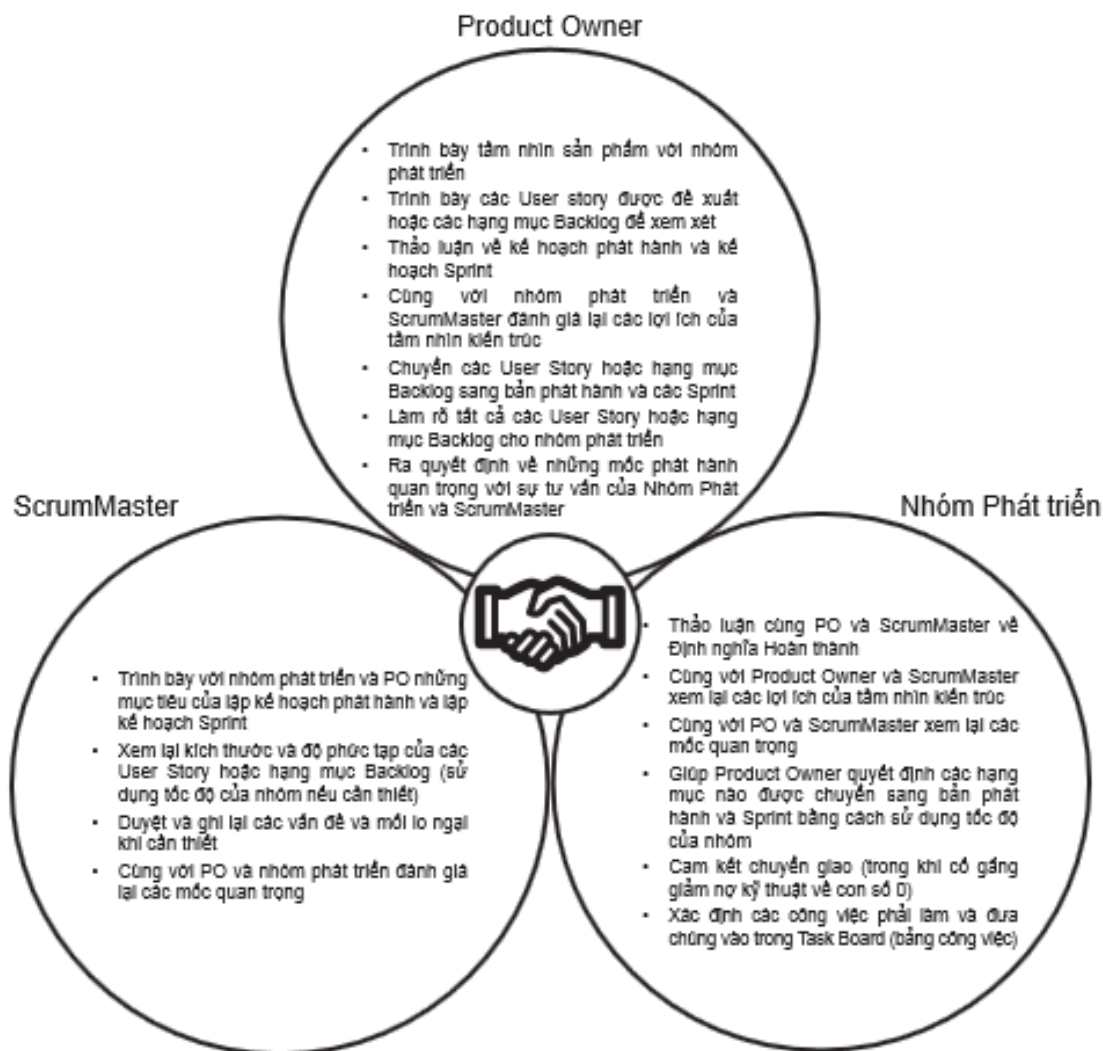


Các thành phần dữ liệu của vòng tròn tiếp theo

# **Từ Phát triển Phần mềm Tăng trưởng đến Phát triển Phần mềm Song song**

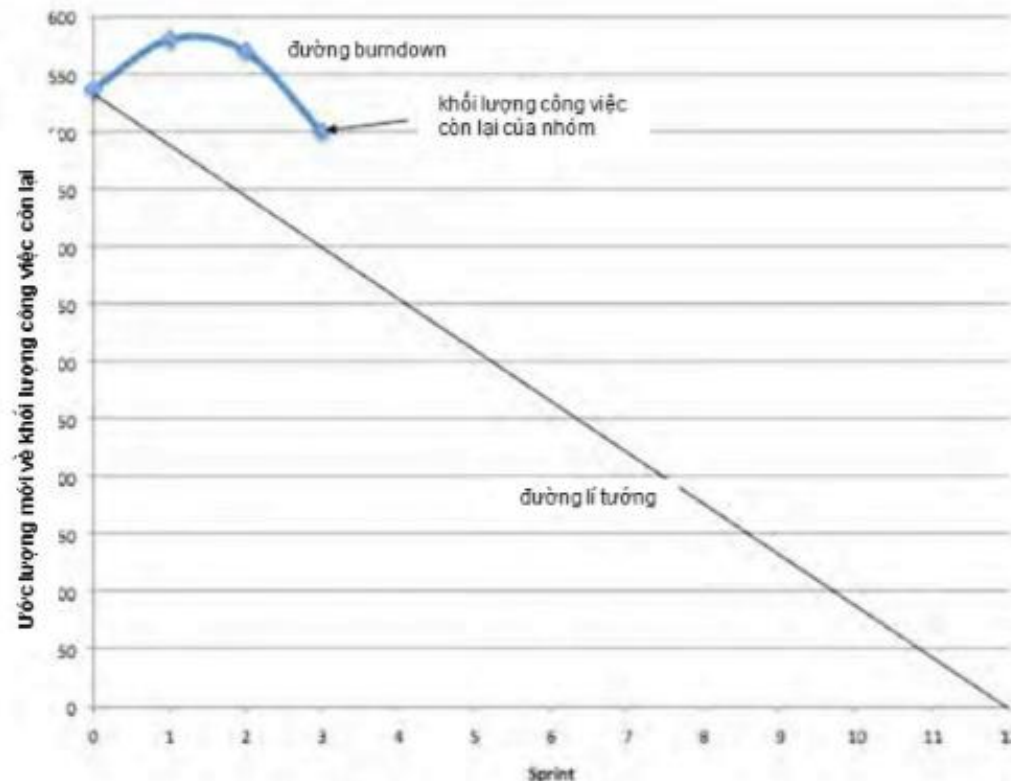
- Ngoài ba câu hỏi nổi tiếng được đưa ra trong những cuộc họp là
  - Bạn đã làm gì ngày hôm qua?
  - Bạn dự định làm gì ngày mai?
  - Có điều gì gây khó khăn cho bạn trong quá trình triển khai hay không?
- Bạn nên thêm hai câu hỏi nữa, chúng sẽ giúp các nhóm con có thể đồng bộ hóa một cách dễ dàng
  - Bạn đang làm gì mà có lợi cho nhóm con của tôi không?
  - Bạn đang làm gì mà gây chậm tốc độ cho nhóm của tôi không?

# Cộng tác trong việc lập kế hoạch phát hành và lập kế hoạch Sprint



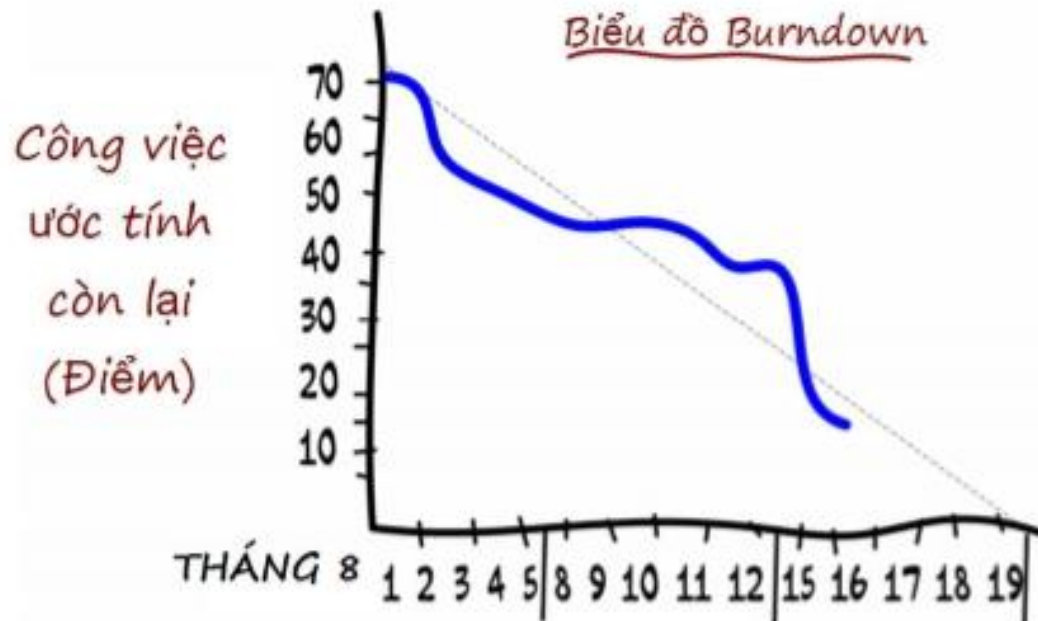
# Biểu đồ Burndown

- Thể hiện **tiến độ** hướng tới Mục tiêu Sprint
- Điều **quan trọng** là còn bao nhiêu việc phải làm để hoàn tất công việc, chứ không phải là đã bỏ ra bao nhiêu công sức trong quá khứ
- Cập nhật lại Burndown hàng ngày (ngay sau họp Scrum hàng ngày)



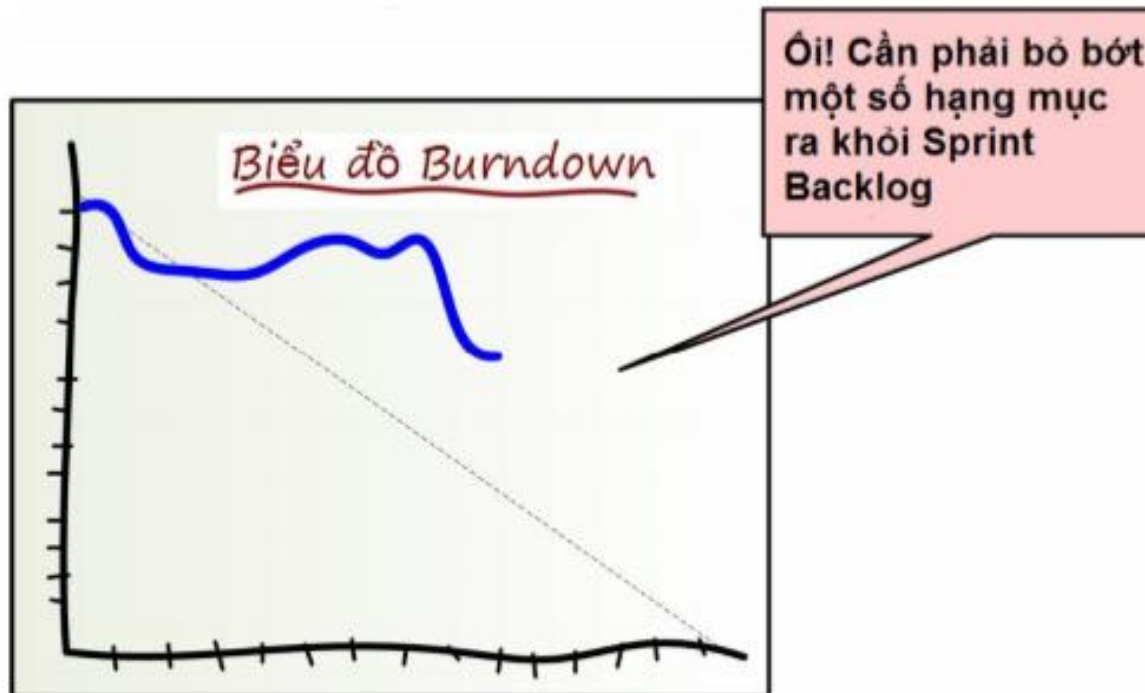
# Biểu đồ Burndown

- Trong ngày đầu tiên của Sprint, ngày 1/8, Nhóm đã ước tính có khoảng 70 điểm (story point) cần phải thực hiện. Điều này có ảnh hưởng đến tốc độ ước tính của toàn bộ Sprint
- Vào ngày 16/8 Nhóm ước tính có khoảng 15 điểm còn lại cần phải làm. Đường thẳng với nét đứt nằm ở trên cho thấy họ đang trong tầm kiểm soát

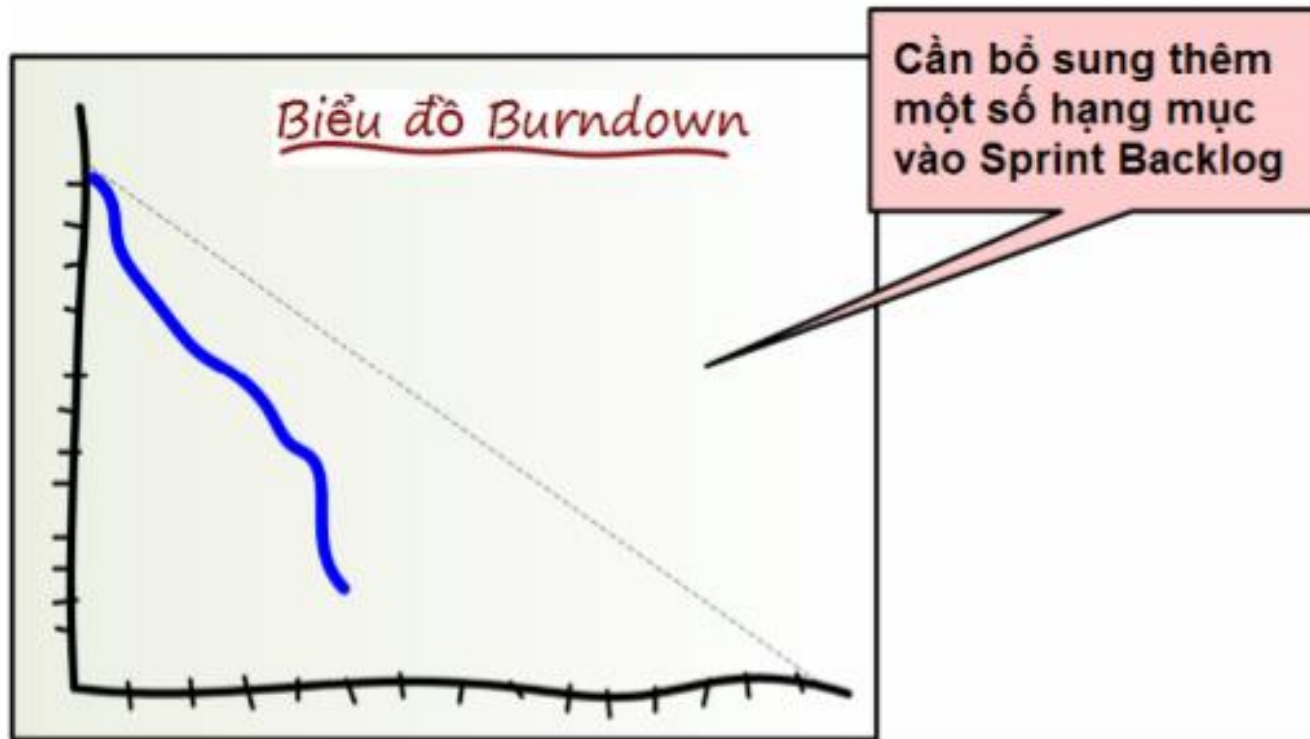


# Những dấu hiệu cảnh báo trên bảng công việc

- Nhìn lướt nhanh bảng công việc sẽ giúp bất kỳ ai có thể thấy được Sprint đang diễn ra như thế nào. Scrum Master có trách nhiệm đảm bảo rằng Nhóm hành động dựa trên những dấu hiệu cảnh báo giống như sau



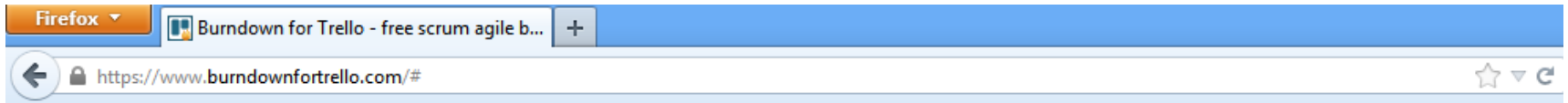
# Những dấu hiệu cảnh báo trên bảng công việc





# Tạo biểu đồ Burndown với trello

- Sử dụng công cụ <https://www.burndownfortrello.com>
- Click nút Connect to Trello để cấp quyền cho ứng dụng sử dụng Trello API

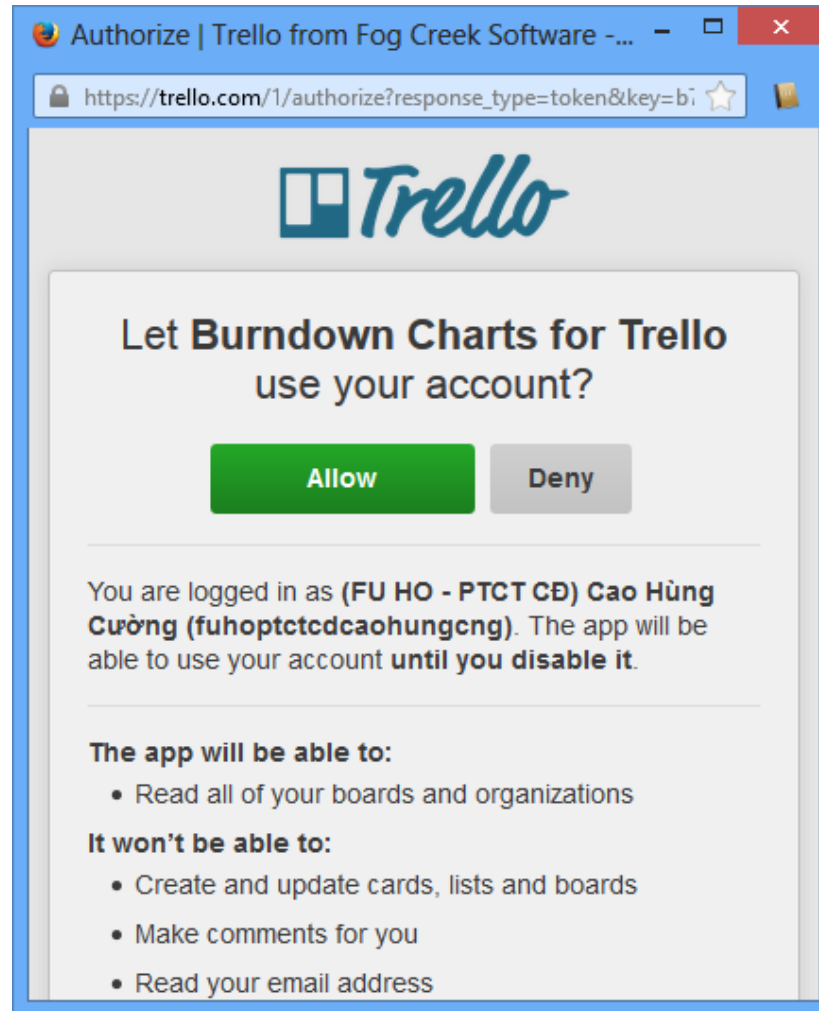


To use this app, please click to authorize this app via the Trello API:

**Connect To Trello**

# Tạo biểu đồ Burndown với trello

- Click nút Allow để cấp quyền cho ứng dụng



# Tạo biểu đồ Burndown với trello

- Click vào Board và biểu đồ burndown hiển thị ở bên phải

Firefox

Burndown for Trello - free scrum agile b...

+

←

https://www.burndownfortrello.com/#

☆ ↻ ↺

Google


🔍

⌵

⌵

🏠


📁

Burndown

for TRELLO

Logged in as (FU H0 - PTCT CĐ) Cao Hùng Cường

[Quick Start](#) [Help / Docs](#) [Account](#)

What's New

[hide](#)

Fixed the 'edit starting hours' form to refresh the board when it finishes updating via the extension.

[see all changes](#) | [upgrade](#)

Welcome Board

Welcome Board

[board settings](#) | [share](#)

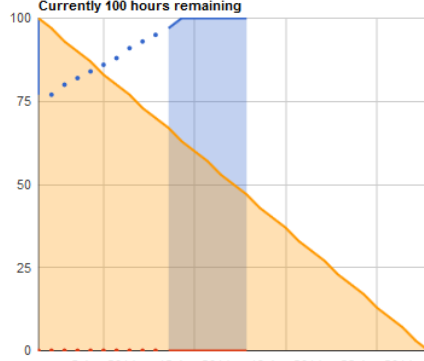
Loaded directly from the Trello API

Reload from API

Reload from Database

Burndown Chart

Currently 100 hours remaining



■ Hours Remaining

■ Hours Completed


■ Ideal Burndown


Summary stats

Total Cards:	21
Remaining Cards:	17
Done Cards:	4
Percent of cards done:	19%
Hours at start:	100 <a href="#">(edit)</a>
Hours est total:	100
Hours remaining:	100
Hours done:	0
Percent of hours done:	0%
----	----
Days Elapsed	15
Daily Burndown	0
Est. Days Left	Infinite
Est. Completion Date	Never

Date-Range

If this Trello Board represents a Sprint or a Product Backlog, adding a start and end date will allow the x-axis to be more accurate and a desired-velocity line to be drawn.

Start Date: 2014-04-01 

End Date: 2014-04-30 

Update Date Range

# Tốc độ nhóm (velocity)

- Khối lượng công việc được hoàn tất trong một Sprint hoặc trên 1 tuần
- Căn cứ để Product Owner tiên đoán thời điểm phát hành
- Căn cứ để Nhóm Phát triển xác định công việc khối lượng cho Sprint
- Trong cùng sản phẩm, với chất lượng đảm bảo, tốc độ tăng có thể phản ánh năng suất lao động
- Các yếu tố ảnh hưởng: độ xác định, thay đổi công nghệ, tăng trưởng kĩ năng, động lực, cam kết, ...

# Họp Scrum hàng ngày

- Duy trì mỗi ngày 15 phút để đồng bộ hóa công việc và lập kế hoạch tức thời
- Một cơ chế “thanh tra và thích nghi” của nhóm tự quản
- Các thành viên trong nhóm sẽ báo cáo cho nhau, không báo cáo cho Scrum Master
- Cố định thời gian, địa điểm; nên đứng khi họp.



# Họp Scrum hàng ngày

- 3 câu hỏi:
  - Đã làm được gì kể từ lần họp trước?
  - Sẽ làm gì từ giờ cho tới lần họp tiếp theo?
  - Có khó khăn gì trong công việc?



## Thảo luận trao đổi tình huống

- Có những khó khăn nào cản trở thực hiện Scrum?



# Workshop 4

- Chuẩn bị trước buổi Workshop:
  - Các nhóm thực hiện trước cuộc họp Scrum hàng ngày (họp đứng trong vòng 20 phút). Lưu thông tin cuộc họp và biên bản cuộc họp lên LMS
- Nội dung trong buổi Workshop
  - Các nhóm tiến hành trao đổi về cuộc họp Scrum hàng ngày
  - Trao đổi với giảng viên để rút kinh nghiệm về cuộc họp Scrum hàng ngày



# Tổng kết nội dung bài học

- Từ tầm nhìn kiến trúc đến lập kế hoạch phát hành và lập kế hoạch sprint
- Từ phát triển phần mềm tăng trưởng đến phát triển phần mềm song song
- Biểu đồ Burndown
- Tốc độ nhóm





**KẾT THÚC**