

## QUẢN LÝ DỰ ÁN VỚI PHẦN MỀM AGILE

### **Bài 4: Ảnh hưởng của tầm nhìn kiến trúc phần mềm tới tốc độ của nhóm và chất lượng phần mềm**

## Nội dung bài học

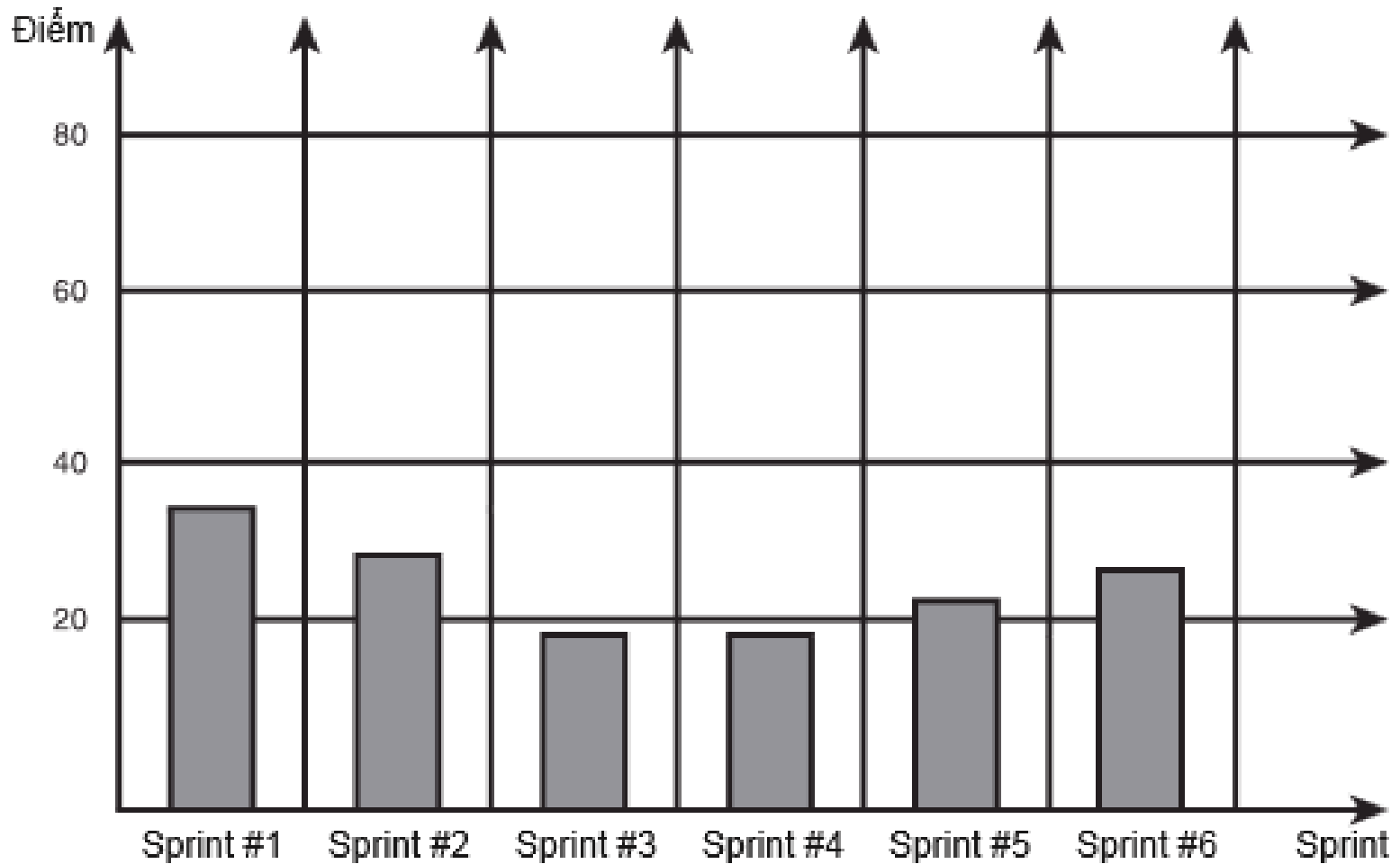
- Tầm quan trọng của Tầm nhìn Kiến trúc
- Làm thế nào để xác định Tầm nhìn Kiến trúc
- Lợi ích khác của việc có Tầm nhìn Kiến trúc
- Phần mềm quản lý dự án Scrum



# **Ảnh hưởng của tầm nhìn kiến trúc**

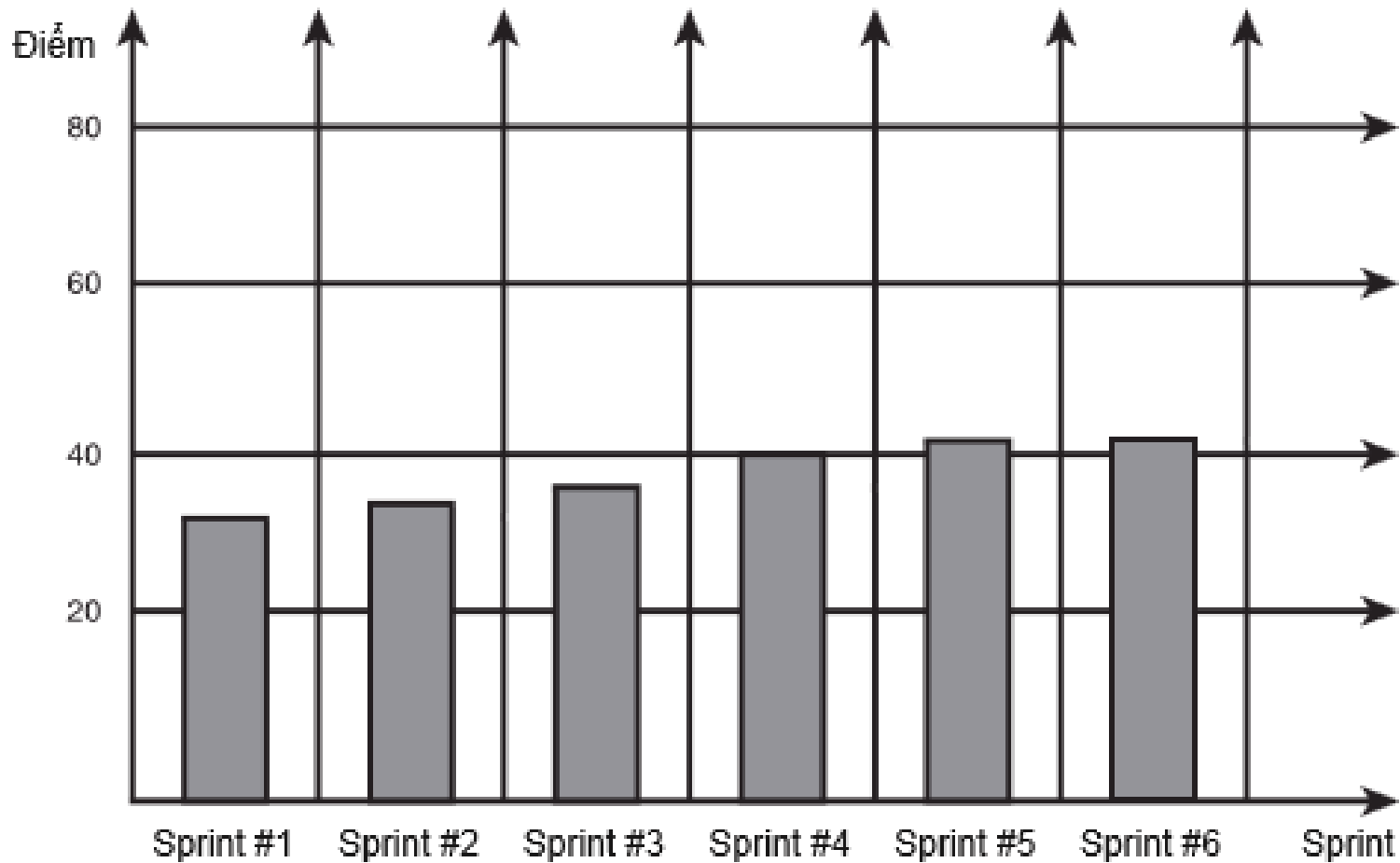
- Trong các dự án Scrum, tốc độ nhóm (số lượng user story mà nhóm dự án có thể bàn giao trong mỗi Sprint) thường dao động theo thời gian
- Dự án càng phức tạp thì tốc độ nhóm càng dao động, điều này thường dẫn tới giảm tốc độ nhóm và ảnh hưởng tới khả năng bàn giao của họ
- Vấn đề này thường có nhiều lý do, một trong những lý do chính, từ góc độ kiến trúc, là nhóm thường bận tâm để hình dung hoặc chỉnh sửa kiến trúc phần mềm trong khi chạy dự án

# Ảnh hưởng của tầm nhìn kiến trúc



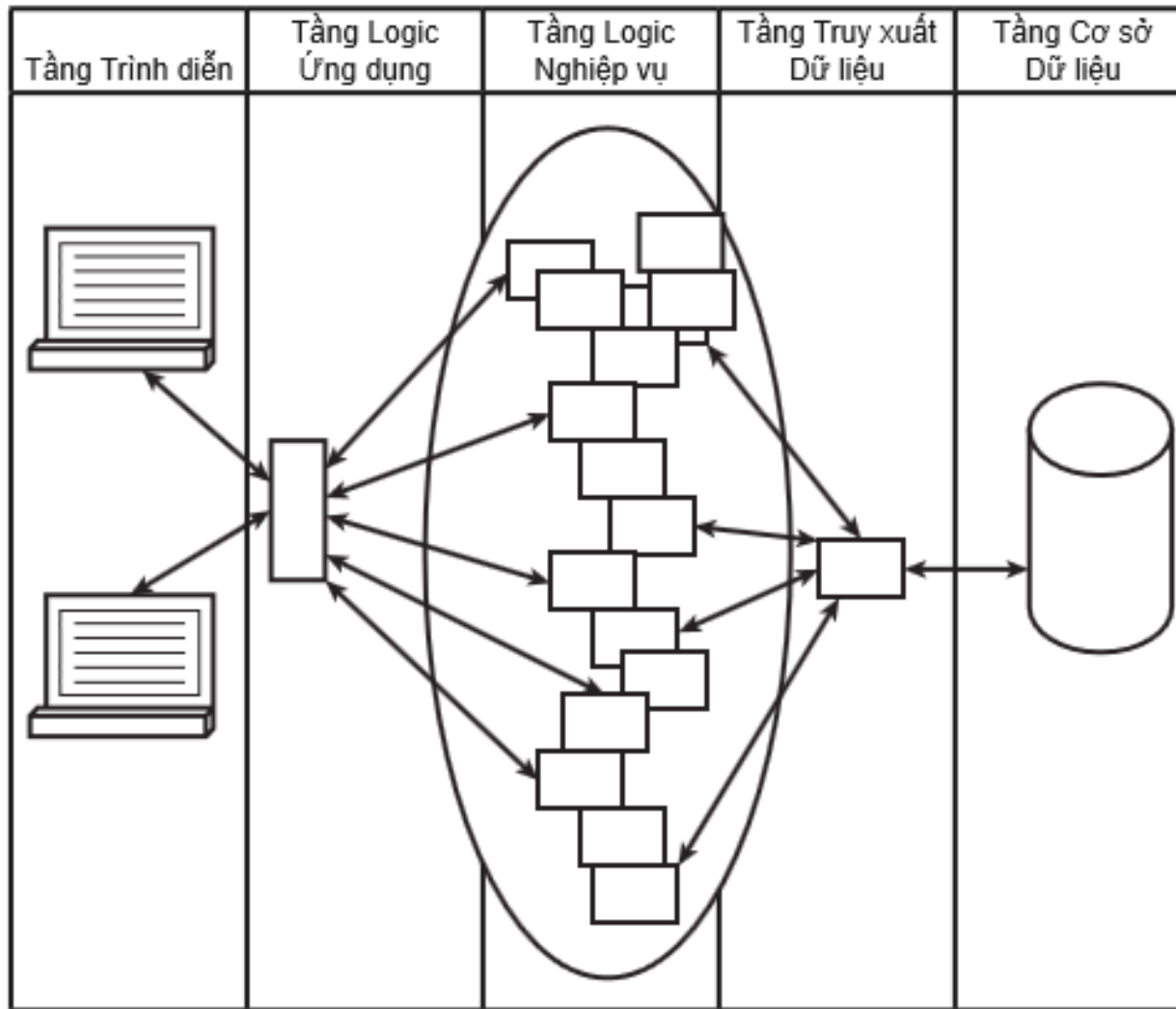
Sự dao động của tốc độ khi thiếu ý đồ kiến trúc

# Tầm quan trọng của tầm nhìn kiến trúc



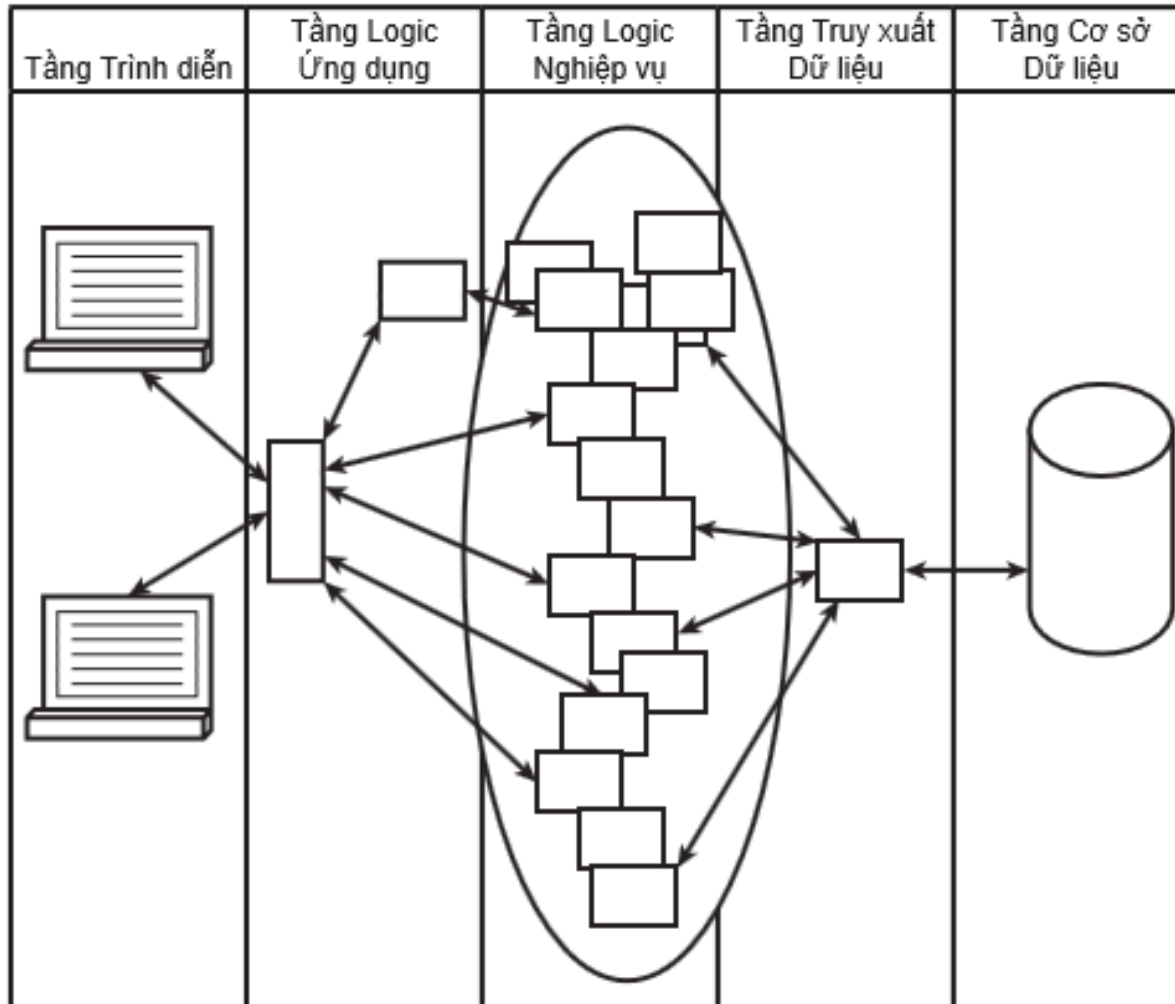
Tốc độ tăng và nhất quán hơn khi có ý đồ kiến trúc

# Thời gian và công sức mà nhóm cần bỏ ra khi không có tầm nhìn kiến trúc



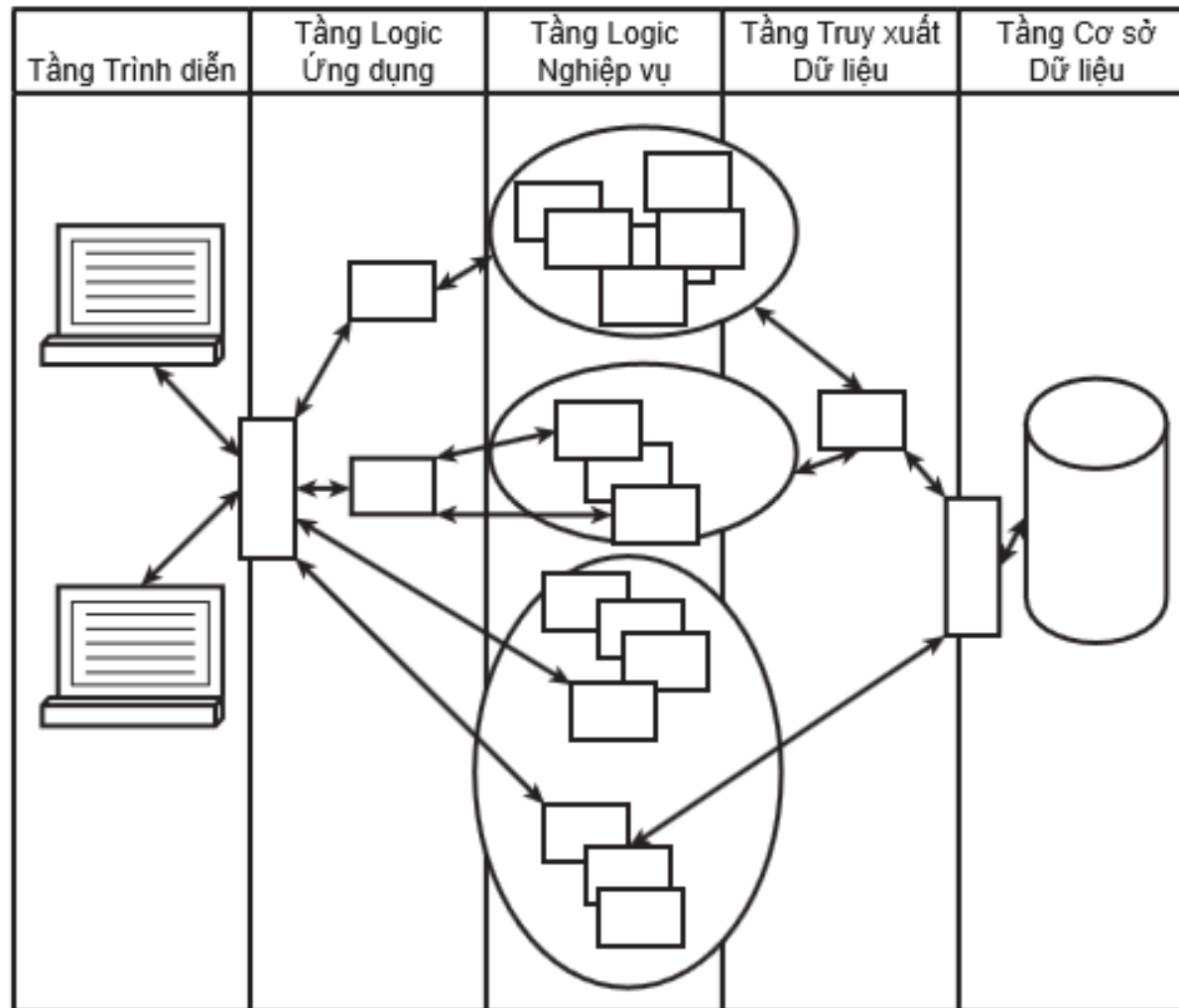
Viết mã ứng dụng mà không có kế hoạch hoặc ý đồ về kiến trúc

# Thời gian và công sức mà nhóm cần bỏ ra khi không có tầm nhìn kiến trúc



Tìm ra kiến trúc trong quá trình triển khai Sprint 2

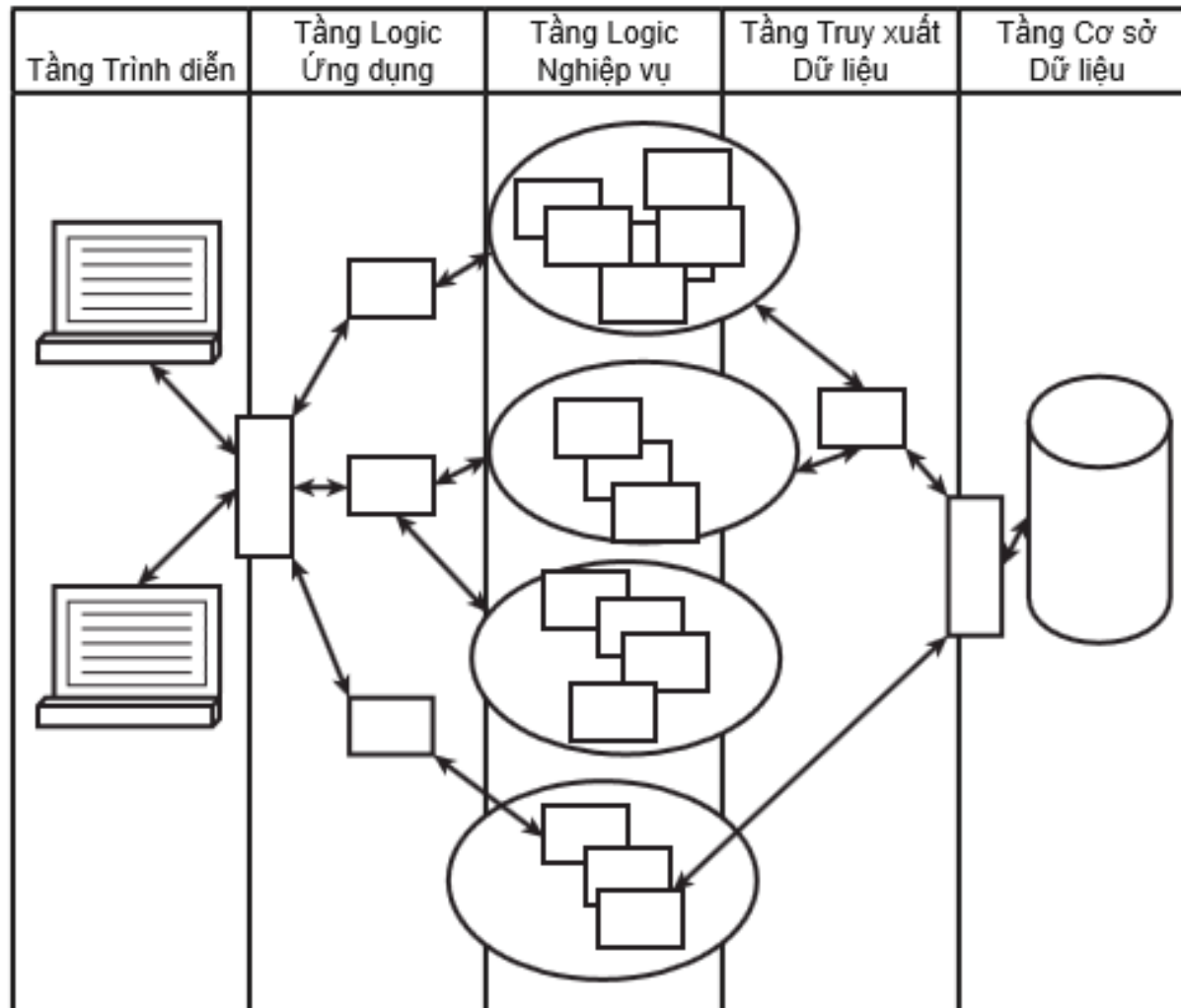
# Thời gian và công sức mà nhóm cần bỏ ra khi không có tầm nhìn kiến trúc



Tìm ra kiến trúc trong Sprint 3



# Thời gian và công sức mà nhóm cần bỏ ra khi không có tầm nhìn kiến trúc

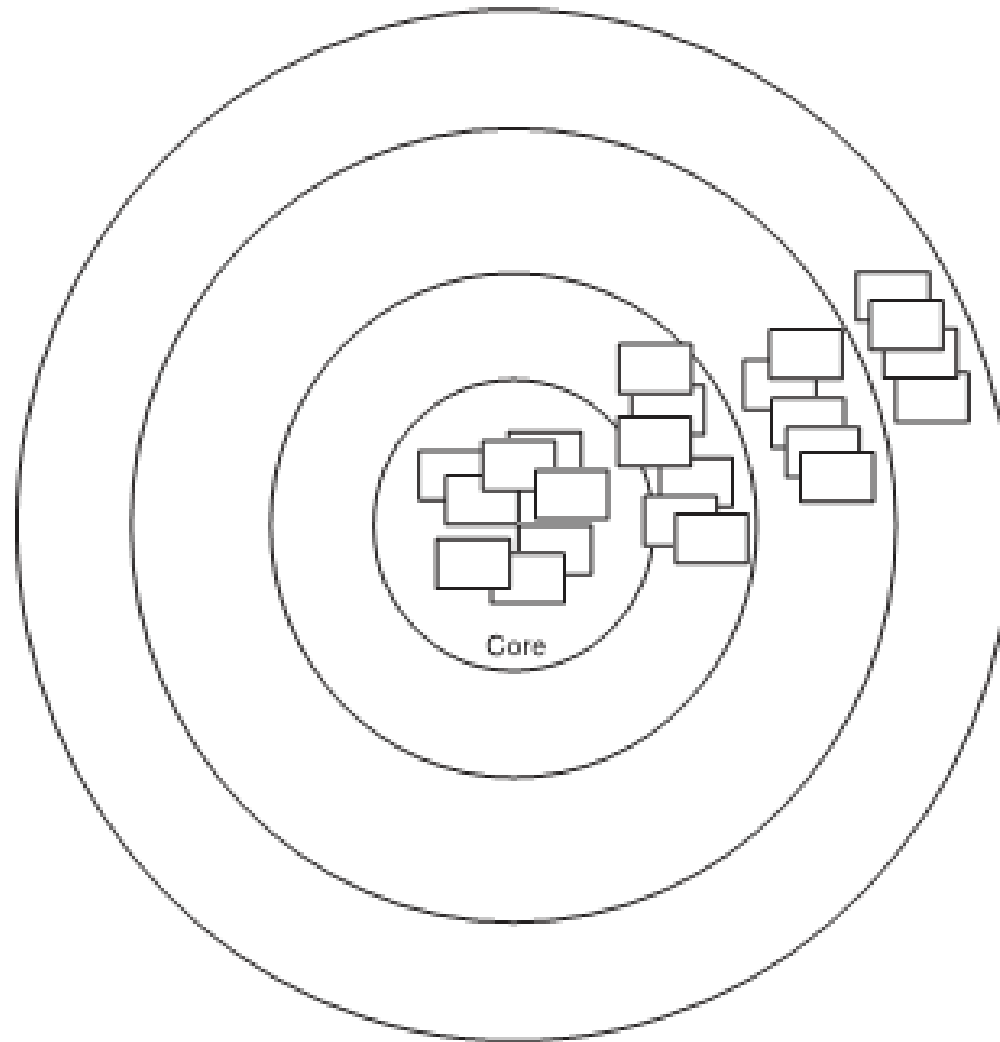


Tìm ra kiến trúc trong Sprint 4

# **Làm thế nào để xác định Tầm nhìn Kiến trúc**

- Nhóm có thể phải thực hiện hai điều để đi đến tầm nhìn kiến trúc. Một trong số đó là xem lại tầm nhìn và mục tiêu của sản phẩm để xác định các dữ liệu kinh doanh chính
- Một cách khác là sử dụng kỹ thuật thu thập yêu cầu bậc cao trực quan, thu thập yêu cầu cho Product Backlog
- Theo chiến lược này, nhóm sẽ có thể cần xác định những user story có thể nhóm lại dựa trên các nghiệp vụ, nhờ đó mà có được các thành phần dữ liệu chung, ổn định

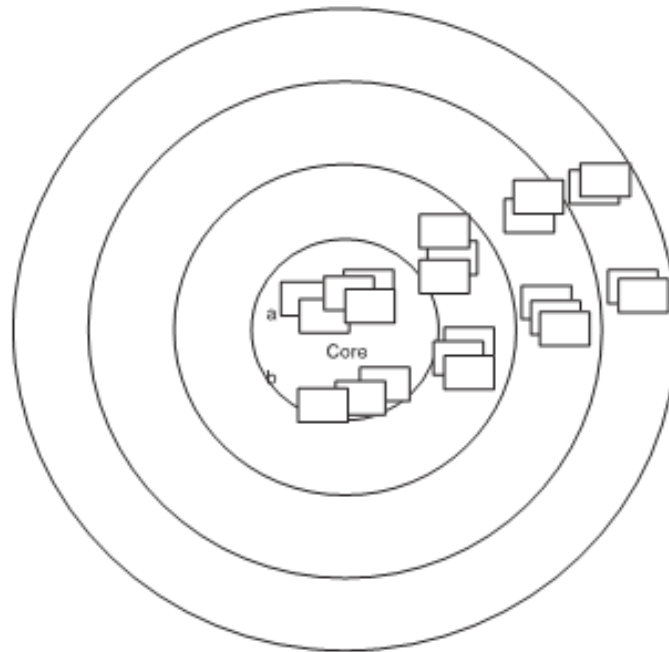
# Làm thế nào để xác định Tầm nhìn Kiến trúc



Phân nhóm các User Story/PBI dựa trên việc xác định dữ liệu chung

# Làm thế nào để xác định Tầm nhìn Kiến trúc

- Nhóm sẽ muốn phân tách nhiều hơn nữa những story sau lần phân nhóm đầu tiên này, để phân tách những story được tạo ra từ những story sẽ được cập nhật, hoặc xóa bỏ



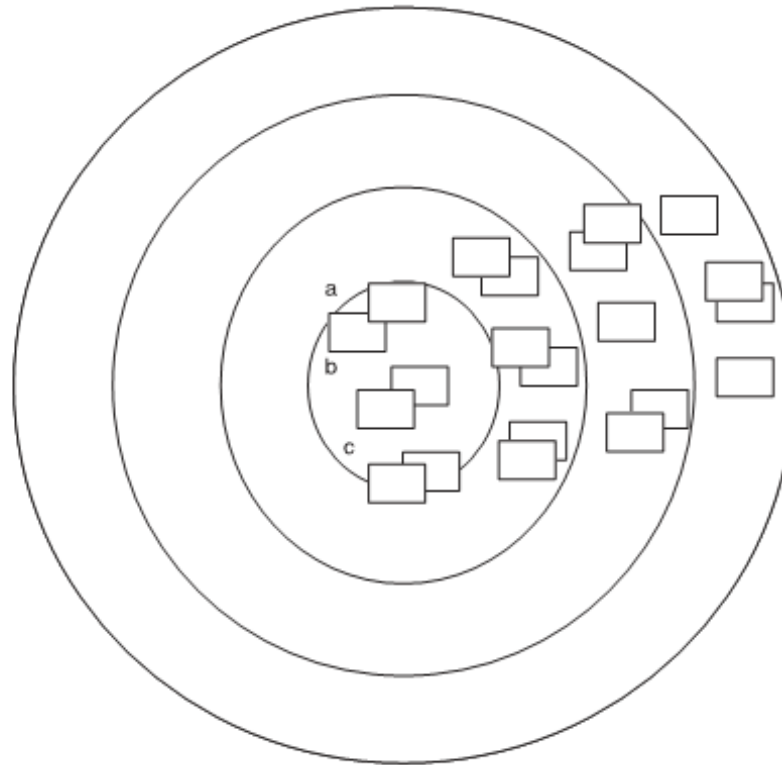
Xác định cụm User story\PBI dựa trên dữ liệu chung

## **Lợi ích khác của việc có tầm nhìn kiến trúc**

- Bên cạnh những lợi ích kể trên, việc xác định sớm dữ liệu nghiệp vụ chung cũng có thể mang lại thêm lợi ích cho nhóm
- Giúp nhóm đặt ra một kiến trúc dữ liệu tốt và ổn định
- Đặc biệt là sẽ tốt cho việc đưa ra báo cáo kho dữ liệu. Trừ khi công ty bạn chỉ sử dụng một ứng dụng, còn không tầm nhìn kiến trúc sẽ giúp ứng dụng của bạn dễ dàng phù hợp kiến trúc dữ liệu doanh nghiệp cùng với các ứng dụng còn lại của công ty

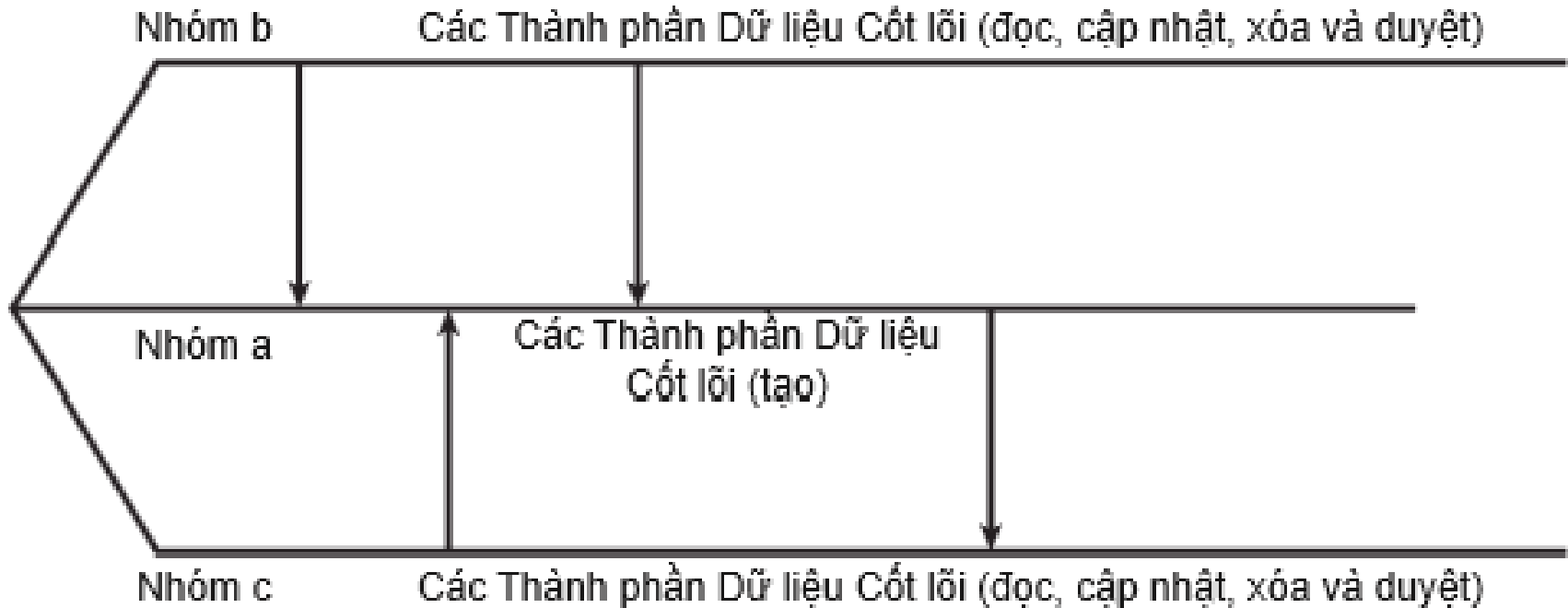
## Lợi ích khác của việc có tầm nhìn kiến trúc

- Có thể sử dụng thêm một bước nữa để phân chia các User story



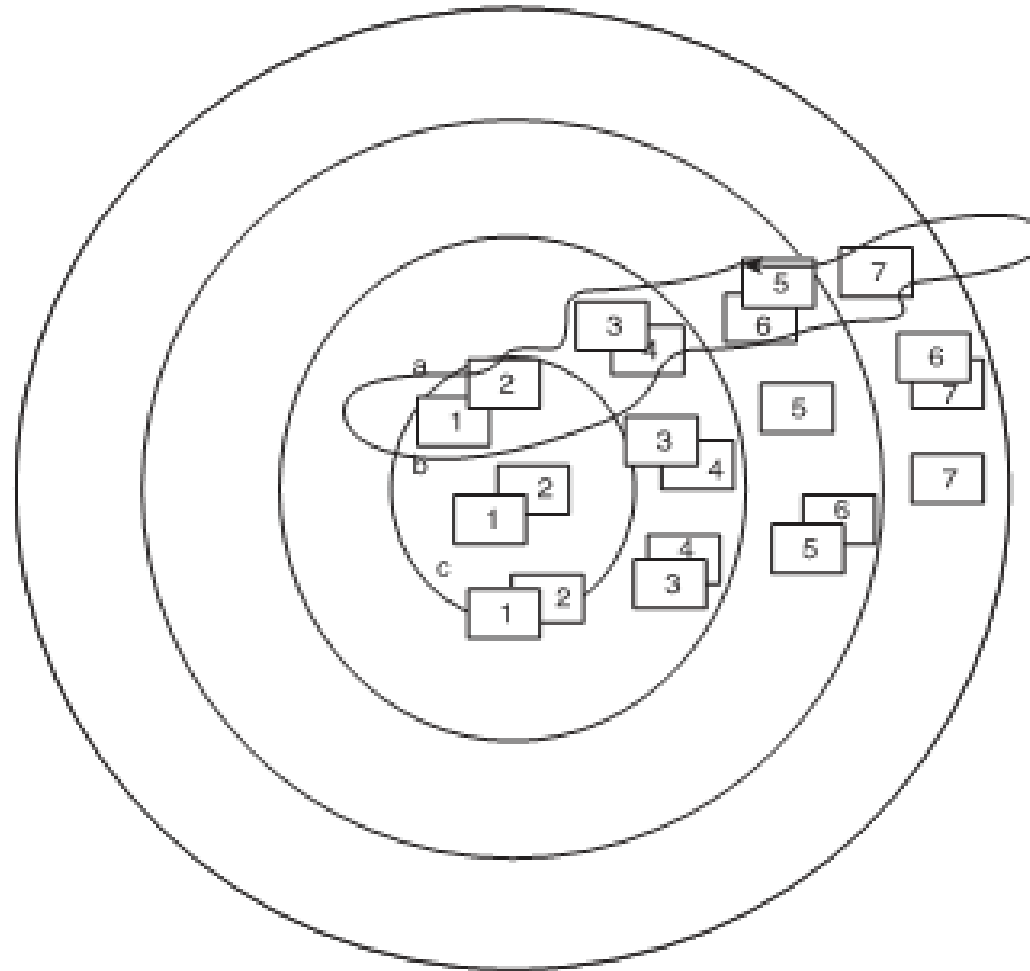
Phân chia các user story dựa trên dữ liệu nghiệp vụ chung thành những cụm tính năng nhỏ hơn nữa

# Lợi ích khác của việc có tầm nhìn kiến trúc



Triển khai đồng thời các User Story/PBI

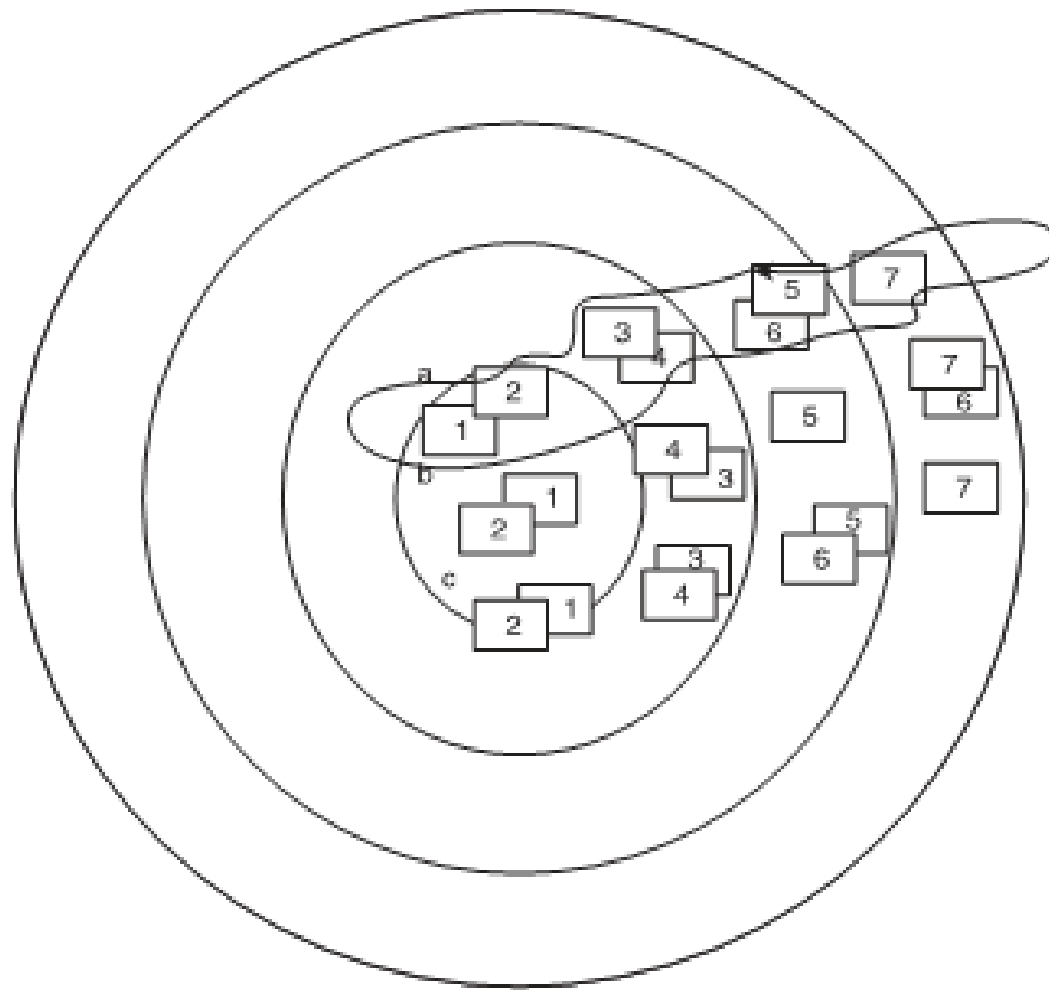
## Lợi ích khác của việc có tầm nhìn kiến trúc



## Thay đổi độ ưu tiên của sản phẩm

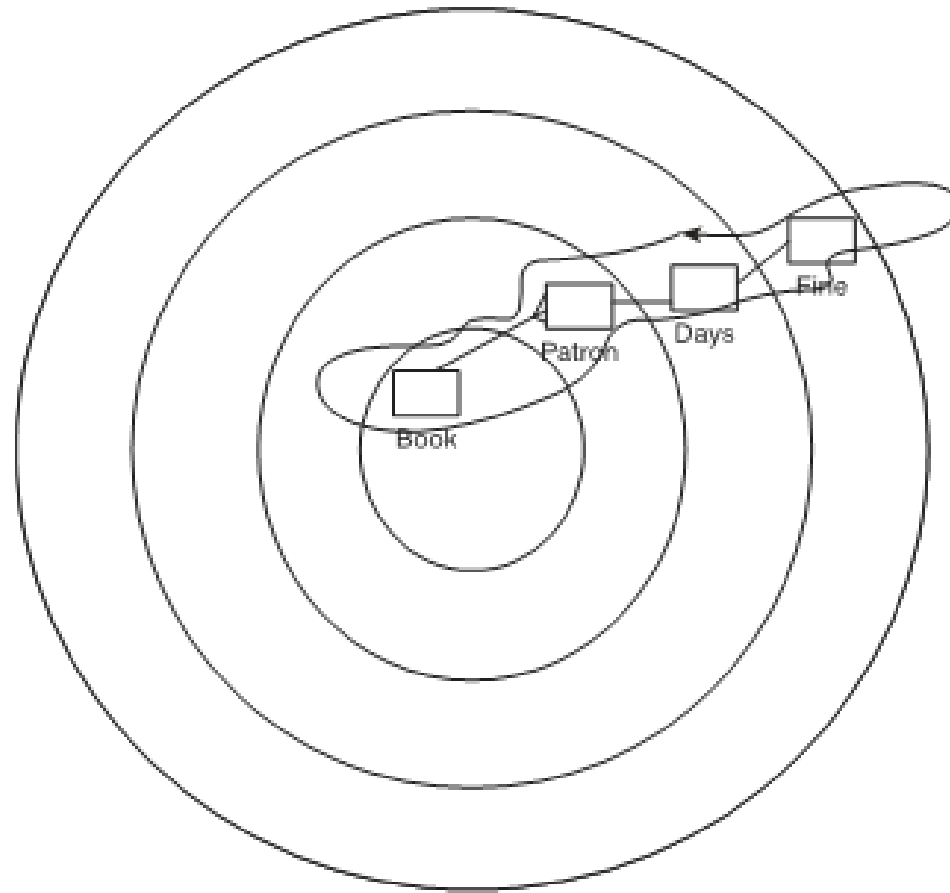


## Lợi ích khác của việc có tầm nhìn kiến trúc



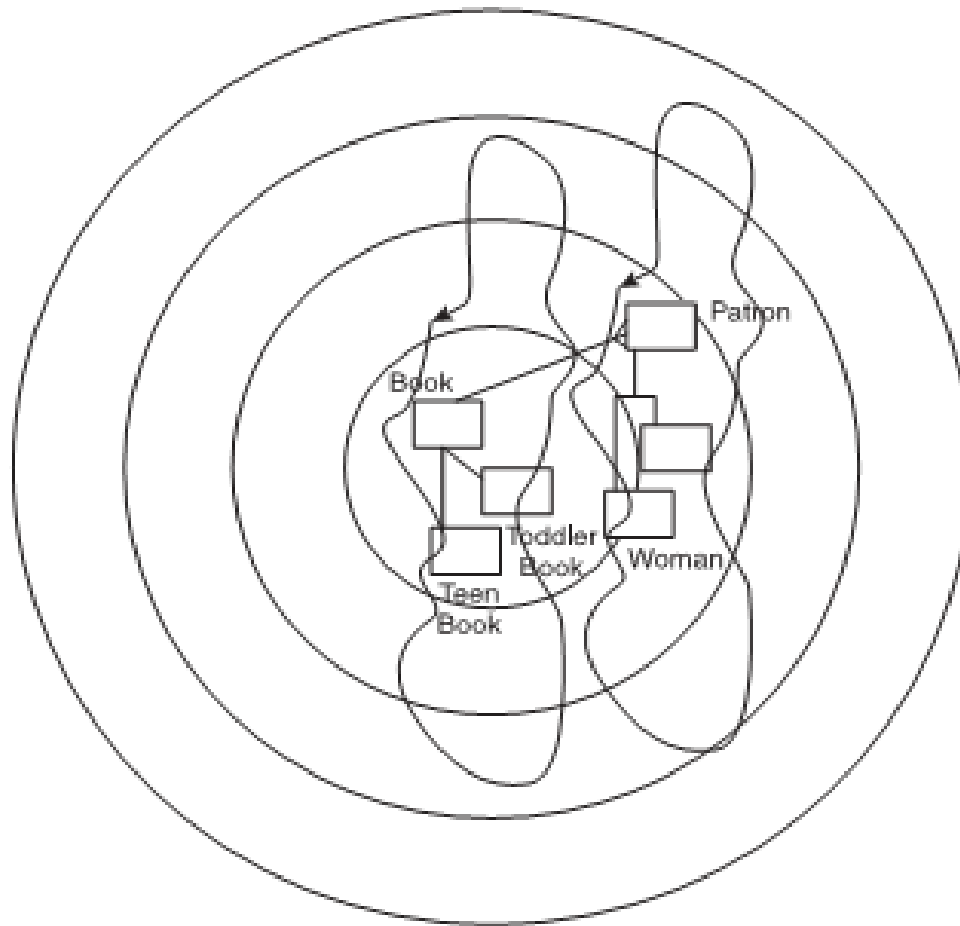
Thay đổi độ ưu tiên của sản phẩm mà không ảnh hưởng đến nhóm

# Lợi ích khác của việc có tầm nhìn kiến trúc



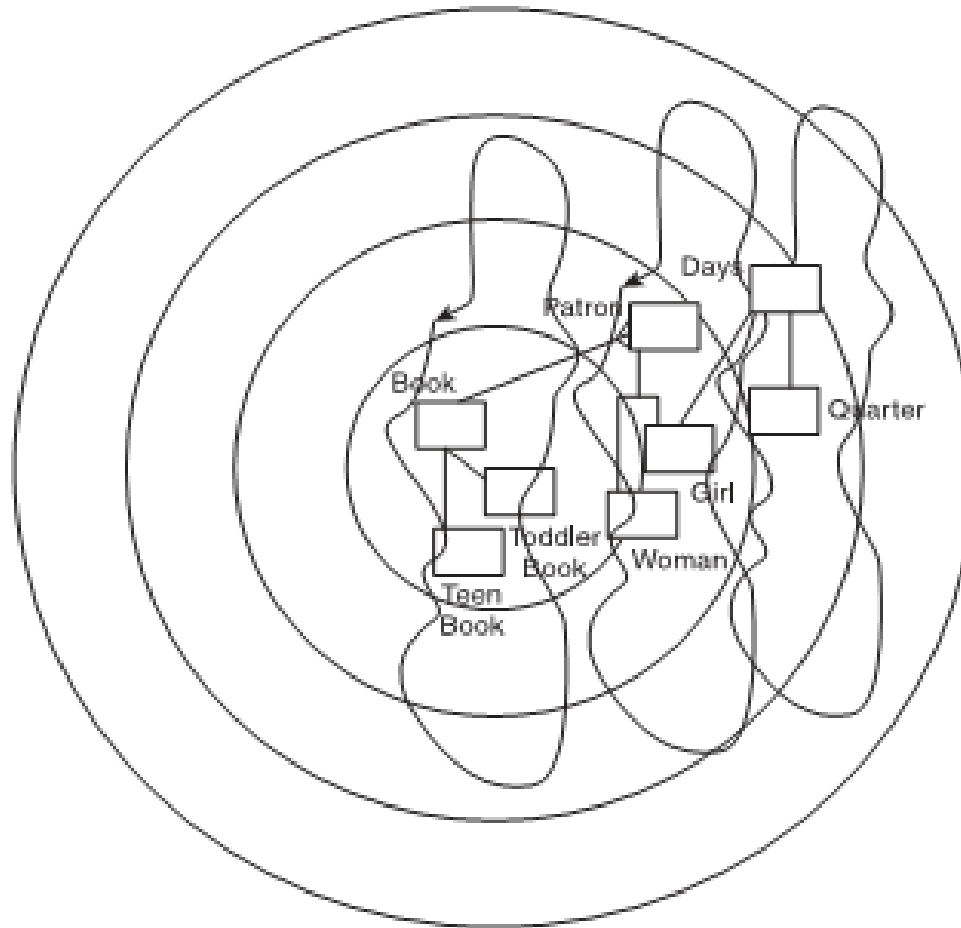
Quá trình bắt đầu một kiến trúc dữ liệu

# Lợi ích khác của việc có tầm nhìn kiến trúc



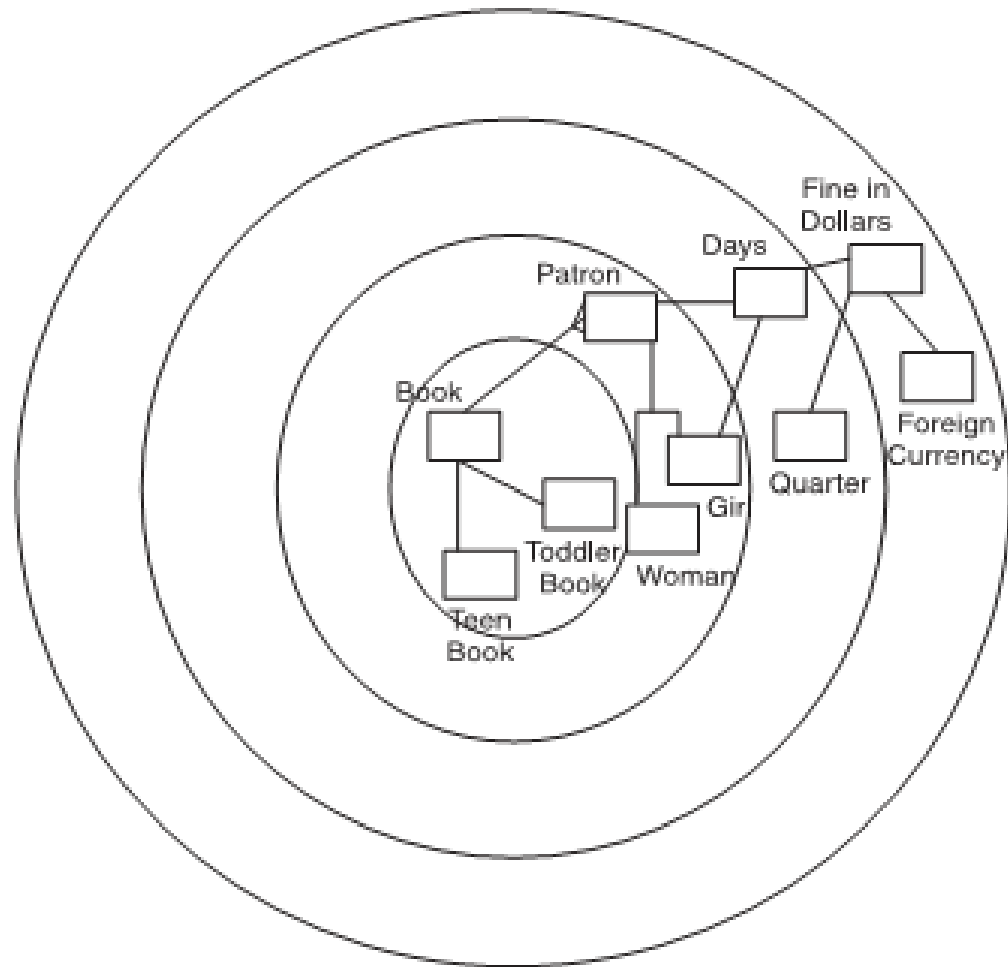
Quá trình kiến trúc dữ liệu theo chiều dọc

# Lợi ích khác của việc có tầm nhìn kiến trúc



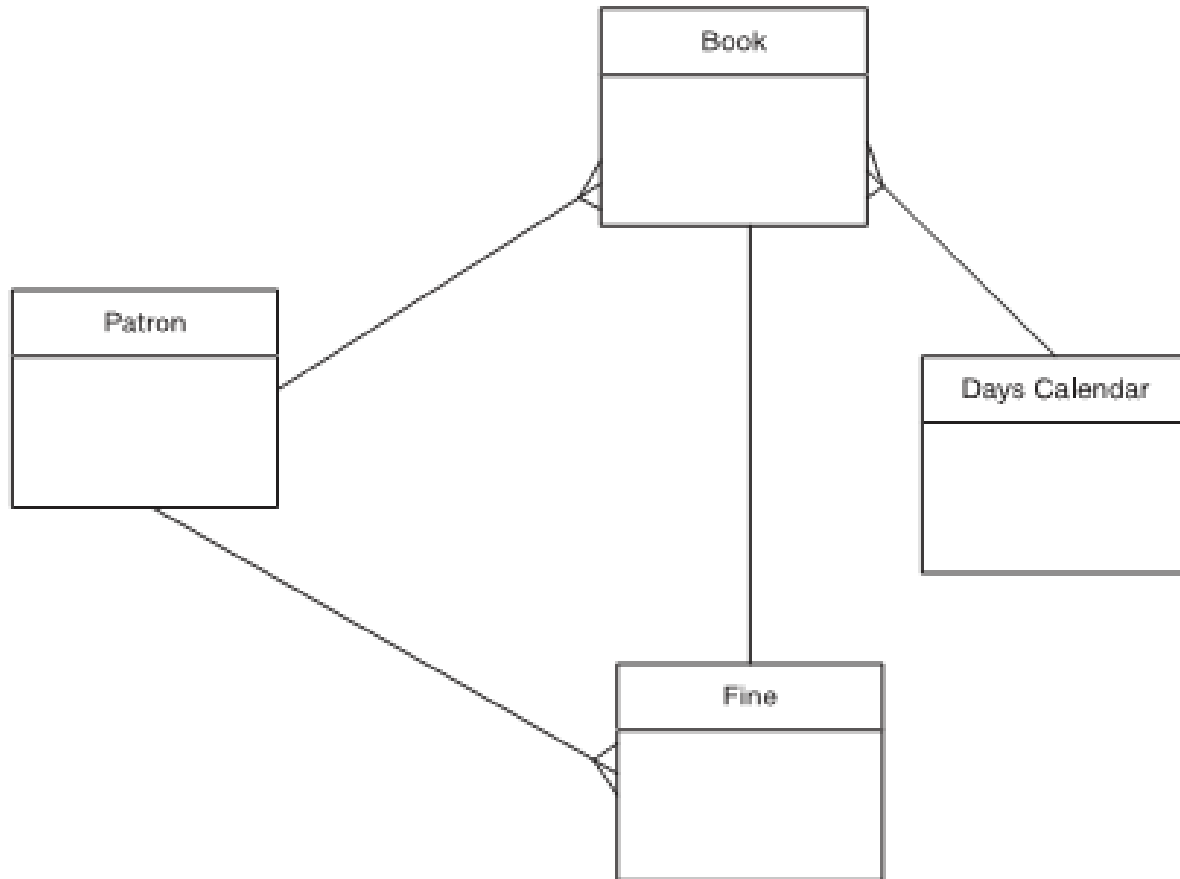
Quá trình kiến trúc dữ liệu

# Lợi ích khác của việc có tầm nhìn kiến trúc



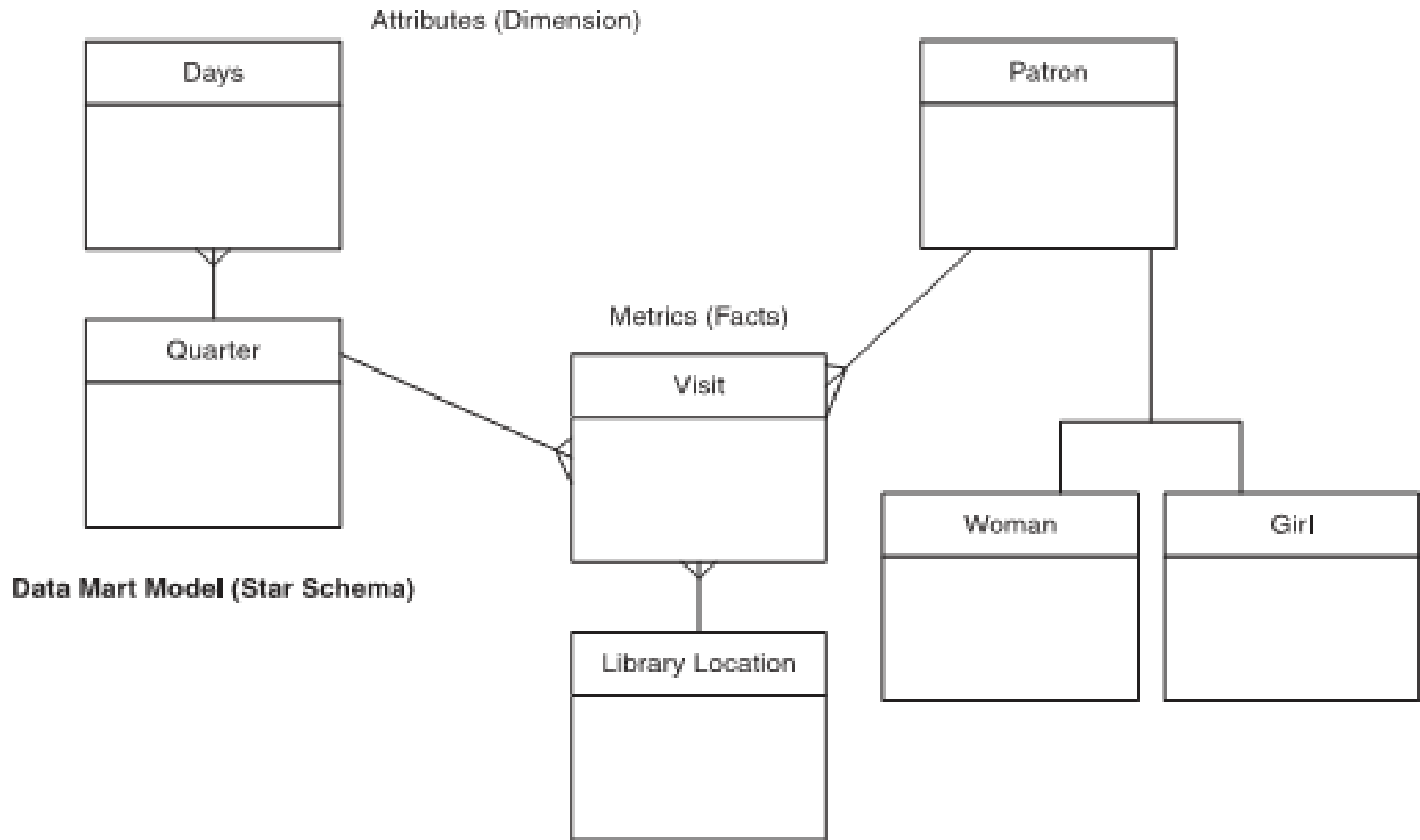
Kiến trúc dữ liệu cuối cùng của ứng dụng

# Lợi ích khác của việc có tầm nhìn kiến trúc



Mô hình dữ liệu giao dịch

# Lợi ích khác của việc có tầm nhìn kiến trúc



Lược đồ hình sao dữ liệu

## Phần mềm quản lý dự án Scrum

- Có một số phần mềm quản lý dự án Scrum
  - Trello (miễn phí)
  - Jira (có phí)
  - PivotalTracker (có phí)



PivotalTracker



# **Giới thiệu phần mềm Trello**

- Trello là một website (tại địa chỉ [trello.com](https://trello.com))
- Công cụ trực tuyến quản lý công việc của cá nhân hay nhóm
- Công việc được chia làm 3 loại:
  - Phải làm (todo)
  - Đang làm (doing)
  - Hoàn thành (done)
- Khi bạn làm chỉ cần quan tâm tới công việc trong trạng thái đang làm, qua đó giúp bạn tập trung xử lý công việc hiệu quả hơn

## **Lợi ích của Trello**

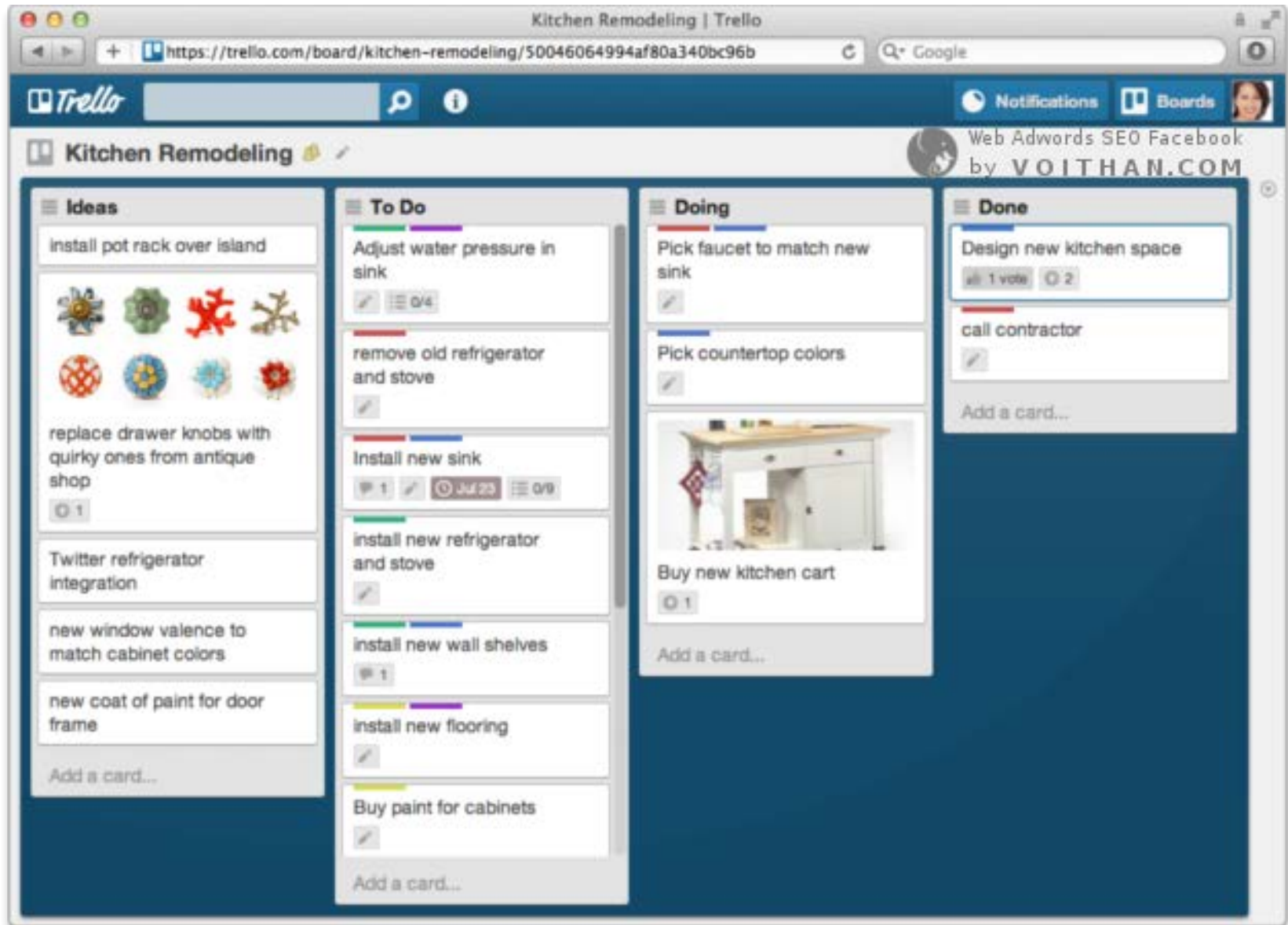
- Không bị quên việc
- Không bị rối
- Sắp xếp công việc ưu tiên
- Kiểm tra công việc

## Đăng ký và sử dụng Trello

- Đăng ký một tài khoản trello (có thể sử dụng gmail)
- Tạo một board quản lý công việc
- Trong bảng quản lý công việc lúc mới tạo có 3 danh sách: todo, doing và done. Đây là nơi chứa toàn bộ đầu mục các công việc của bạn để thực hiện



# Đăng ký và sử dụng Trello



# Tạo các danh sách công việc trên Trello

- Mặc định Trello tạo sẵn 3 danh sách công việc: Todo, Doing và Done. Bạn có đổi tên cho phù hợp với cách hiểu của bạn, hoặc có thể thêm cột tùy ý
  - **Todo (việc phải làm):** Tại đây bạn tạo các đầu việc (Add a card) cần làm
  - **Doing (đang làm):** Danh sách này là các việc bạn đang thực hiện. Tốt nhất là chỉ có 1 đầu việc trong mục này, vì cùng 1 lúc bạn chỉ nên thực hiện 1 công việc thôi
  - **Done (hoàn thành):** Các công việc đã thực hiện xong sẽ được để ở đây

## Quản lý công việc với Trello

- Khi bắt đầu một ngày làm việc bạn sẽ bắt đầu ở cột Todo, viết hết các công việc cần làm ở cột này, có thể bổ sung trong cả ngày
- Sau đó bạn chọn việc phù hợp từ cột Todo kéo sang cột Doing và tiến hành công việc đó.
- Làm xong công việc ở cột Doing bạn kéo công việc sang cột Done, vậy là bạn đã hoàn thành một công việc.
- Bạn tiếp tục tìm công việc từ cột Todo và kéo sang cột Doing và cứ tiếp tục như thế cho tới khi hết công việc tại cột Todo

# Thảo luận trao đổi tình huống

- Cập nhật bảng công việc như thế nào?
- Ứng xử với những người đến muộn?
- Ứng xử với những câu trả lời "Tôi không biết làm gì ngày hôm nay"?



# Workshop 3

- Chuẩn bị trước buổi Workshop:
  - Lưu tài liệu mô tả về user story của Sprint và đánh giá chi tiết về user story lên LMS (sử dụng mẫu cung cấp trên LMS)
  - Lưu thông tin cuộc họp hàng ngày (dưới dạng file word) lên LMS
- Nội dung trong buổi Workshop
  - Thảo luận và trao đổi với giảng viên về các yêu cầu sau:
    - Mô tả yêu cầu cho sprint
    - Đưa ra danh sách user story của sprint
    - Đánh giá các user story của sprint



# Tổng kết nội dung bài học

- Với một tầm nhìn kiến trúc rõ ràng, tốc độ nhóm không dao động nhiều
- Bằng việc phân tách các user story theo các thành phần dữ liệu nghiệp vụ chung, nhóm sẽ có thể tổ chức công việc của mình và phát triển song song
- Sử dụng công cụ Trello để quản lý dự án





**KẾT THÚC**