

## Bài 1:

### Đoạn Hội Thoại Sprint Review – Phiên Bản Chính Sửa

**Dự án:** Ứng dụng Quản lý Nhóm (Team Management App)

**Sprint Goal:** Hoàn thành chức năng tạo và giao việc cơ bản cho nhóm.

Vai trò	Tên	Lời thoại
<b>Product Owner (PO)</b>	<b>Duy</b>	Chào mọi người, cảm ơn đã tham gia buổi Sprint Review hôm nay. Mục tiêu của Sprint này là hoàn thành tính năng <b>tạo công việc và giao việc cơ bản</b> cho các thành viên trong nhóm. Tính năng này giúp người dùng nhanh chóng tạo đầu việc mới và chỉ định người phụ trách. Bây giờ, mời Dev Tùng trình bày phần demo.
<b>Developer (Dev)</b>	<b>Tùng</b>	Vâng, cảm ơn PO Duy Như mọi người thấy, tôi đang ở màn hình dự án “Marketing Q3”. Tôi sẽ bấm vào nút <b>“Thêm công việc mới”</b> . Pop-up tạo công việc xuất hiện – tôi nhập tên công việc: <b>“Thiết kế banner cho chiến dịch X”</b> , chọn deadline vào cuối tuần này, và chỉ định người phụ trách là <b>Minh</b> . Sau khi nhấn <b>“Lưu”</b> , công việc được thêm ngay vào danh sách và hiển thị đầy đủ tên cũng như người được giao.
<b>Stakeholder</b>	<b>Hải</b>	Cảm ơn phần demo của Tùng. Giao diện khá trực quan và thao tác rất nhanh – tôi đánh giá cao điều đó. Tuy nhiên, tôi có một góp ý nhỏ: ở bước tạo công việc, liệu có thể thêm luôn phần <b>mô tả chi tiết</b> không? Với những đầu việc phức tạp, việc cho phép nhập mô tả từ đầu sẽ tiện hơn thay vì phải mở chỉnh sửa sau này.
<b>PO</b>	<b>An</b>	Cảm ơn anh Hải về góp ý rất giá trị. Việc bổ sung trường mô tả ngay khi tạo công việc chắc chắn sẽ cải thiện trải nghiệm người dùng. Chúng tôi sẽ ghi nhận và đưa yêu cầu này vào Product Backlog để thảo luận mức độ ưu tiên cho Sprint tới.

## Bài 2:

### a. Product Backlog Item (PBI)

Tên PBI:

**Bổ sung Bộ lọc Thông kê Hoạt động Theo Ngày**

Mô tả (Role – Goal – Benefit):

- Là một Trưởng phòng Vận hành (Stakeholder),
- Tôi muốn có thể áp dụng các bộ lọc ngày (một ngày cụ thể hoặc 24 giờ gần nhất) trong phần thống kê hoạt động của người dùng,
- Để tôi có thể kiểm tra và phân tích dữ liệu hoạt động một cách chính xác theo từng ngày, hỗ trợ theo dõi hiệu suất và ra quyết định tốt hơn.

(Hoặc dạng Business Requirement – Yêu cầu Nghiệp vụ):

**Business Requirement:**

Hệ thống cần cho phép người dùng lọc và hiển thị thống kê hoạt động theo mốc **một ngày cụ thể** hoặc **24 giờ gần nhất**, thay cho việc chỉ xem dữ liệu tổng hợp theo tuần như hiện tại.

### b. Acceptance Criteria (ACs) – Tiêu chí Chấp nhận

**AC 1 – Bộ lọc Ngày Cụ thể:**

- Người dùng có thể chọn một **ngày cụ thể** (ví dụ: 03/11/2023) để xem thống kê hoạt động.

**AC 2 – Bộ lọc 24 Giờ Gần Nhất:**

- Người dùng có tùy chọn xem thống kê hoạt động trong **24 giờ gần nhất** tính từ thời điểm hiện tại.

**AC 3 – Hiển thị Dữ liệu Chính xác:**

- Khi bộ lọc được áp dụng, biểu đồ và các chỉ số thống kê (VD: số công việc hoàn thành, thời gian hoạt động, mức độ tương tác...) **chỉ hiển thị dữ liệu nằm trong khoảng thời gian đã chọn**, không kèm dữ liệu ngoài phạm vi.

#### **AC 4 – Hiển thị Trạng thái Bộ lọc:**

- Giao diện phải thể hiện rõ ràng khoảng thời gian đang được áp dụng, ví dụ:
  - “Thống kê hoạt động: 03/11/2023”
  - “Thống kê hoạt động: 24 giờ gần nhất”

### **Bài 3:**

**Dự án Giả lập:** Ứng dụng Quản lý Nhóm (Team Management App)

**Sprint Goal:** Hoàn thành chức năng "Tạo Công Việc" (Create Task) và "Bộ Lọc Cơ Bản" (Basic Filter) trên danh sách công việc.

**Thời gian Sprint:** 14 ngày (Từ Ngày 1 đến Ngày 14)a. Timeline Sprint (14 Ngày)

Ngày	Hoạt động Chính
Ngày 1	Sprint Planning, Bắt đầu phát triển chức năng Tạo Công Việc.
Ngày 2	Dev hoàn thành phần backend chức năng Tạo Công Việc.
Ngày 3	<b>Sự kiện 1:</b> Gặp sự cố về môi trường Staging.
Ngày 4	Fix sự cố môi trường. Bắt đầu phát triển UI/UX cho Tạo Công Việc.
Ngày 5	Hoàn thành và đưa chức năng Tạo Công Việc lên môi trường Dev để test nội bộ.
Ngày 6	Bắt đầu phát triển chức năng Bộ Lọc Cơ Bản (theo người giao việc).
Ngày 7	Dev Minh nghỉ phép đột xuất (nửa ngày).
Ngày 8	<b>Sự kiện 2:</b> Phát sinh yêu cầu mới từ Stakeholder.
Ngày 9	Dev Tùng hoàn thành phần backend Bộ Lọc Cơ Bản.
Ngày 10	Kiểm tra và phát hiện <b>Sự kiện 3:</b> Bug nghiêm trọng ở Tạo Công Việc.
Ngày 11	Cả đội tập trung fix Bug và review code.

Ngày 12	<b>Sự kiện 4:</b> PO Quyết định loại bỏ một PBI nhỏ để đảm bảo Sprint Goal.
Ngày 13	Hoàn thành chức năng Bộ Lọc Cơ Bản, sẵn sàng demo.
Ngày 14	<b>Sự kiện 5:</b> Sprint Review và Sprint Retrospective.

### b. 5 Sự Kiện Quan Trọng Trong Sprint

Ngày X	Sự kiện	Mô tả	Ảnh hưởng	Bài học rút ra (1 câu)
Ngày 3	<b>Sự có môi trường Staging</b>	Môi trường Staging (dùng để kiểm thử tích hợp) bị lỗi kết nối cơ sở dữ liệu khiến nhóm QA không thể bắt đầu kiểm thử sớm theo kế hoạch.	<b>Tiêu cực:</b> Làm chậm quá trình QA 1 ngày.	Cần có quy trình kiểm tra và bảo trì định kỳ môi trường kiểm thử.
Ngày 8	<b>Phát sinh yêu cầu mới</b>	Stakeholder yêu cầu thêm trường "Mô tả chi tiết" vào màn hình tạo công việc sau khi thấy bản nháp. Yêu cầu này nằm ngoài Sprint Backlog.	<b>Tiêu cực:</b> Gây xao nhãng và phải họp nhanh để quyết định đưa vào Sprint Backlog sau.	Mọi yêu cầu thay đổi (Scope Creep) phải được xử lý thông qua PO và được đưa vào Product Backlog để đánh giá lại ưu tiên.
Ngày 10	<b>Phát hiện Bug nghiêm trọng</b>	Chức năng tạo công việc bị lỗi không thể gán công việc cho thành viên nếu tên công việc có ký tự đặc biệt, gây crash app.	<b>Tiêu cực:</b> Phải dành thời gian 1 ngày để Dev và QA cùng giải quyết triệt để.	Cần tăng cường unit test (kiểm thử đơn vị) và kiểm thử biên (boundary test) ngay từ đầu để bắt lỗi sớm hơn.
Ngày 12	<b>PO Cắt giảm Scope</b>	PO (Product Owner) quyết định tạm gác lại PBI "Lọc theo trạng thái công việc" (một PBI nhỏ) để tập trung hoàn thành 100% hai PBI chính ("Tạo	<b>Tích cực:</b> Đảm bảo Sprint Goal được hoàn thành và Product Increment có	PO cần linh hoạt trong việc bảo vệ Sprint Goal bằng cách cắt giảm các PBI có ưu tiên thấp hơn khi rủi ro tiến độ tăng cao.

		Công Việc" và "Bộ Lọc Cơ Bản").	thể sử dụng được.	
Ngày 14	Demo thành công	Chức năng "Tạo Công Việc" và "Bộ Lọc Cơ Bản" được demo thành công và nhận được phản hồi tích cực về tốc độ xử lý từ Stakeholder trong buổi Sprint Review.	<b>Tích cực:</b> Đạt được Sprint Goal, tạo ra giá trị gia tăng và nâng cao tinh thần nhóm.	Việc tập trung vào một Sprint Goal rõ ràng giúp cả nhóm đồng lòng và dễ dàng đo lường sự thành công.

#### Bài 4:

Kỹ thuật Retrospective được chọn: Mad – Sad – Glad

Giải thích lý do lựa chọn (5–7 dòng):

- Phù hợp:** Kỹ thuật Mad – Sad – Glad (MSG) phù hợp nhất vì vấn đề cốt lõi của đội hiện tại là **tinh thần và cảm xúc** tiêu cực ("stress", "tinh thần làm việc tụt dốc", "ít giao tiếp"). MSG tạo một không gian an toàn, trực tiếp yêu cầu thành viên chia sẻ những cảm xúc thực tế (bực bội, buồn bã, vui vẻ) để giải tỏa căng thẳng trước.
- Giải quyết vấn đề:** Nó trực tiếp giúp **giải tỏa cảm xúc và khơi gợi sự chia sẻ thật** (Mục tiêu 1 & 2). Những vấn đề được đưa ra dưới mục "Mad" và "Sad" sẽ là mạnh mẽ chính để Scrum Master dẫn dắt team phân tích **nguyên nhân sâu xa (root-cause)** của stress và bug tồn tại cuối Sprint.
- So với các kỹ thuật khác:** Các kỹ thuật như Start – Stop – Continue hoặc 4Ls quá tập trung vào hành động và quy trình, không đủ mạnh để phá vỡ rào cản cảm xúc và giao tiếp đang tồn tại trong team. Timeline có thể giúp tìm nguyên nhân của bug, nhưng khó hiệu quả nếu team không sẵn sàng mở lòng chia sẻ.

#### Bài 5:

**Tính năng được chọn:** Tìm kiếm sản phẩm (Product Search) trên Ứng dụng Thương mại điện tử.

**Dự án:** Ứng dụng Thương mại điện tử XYZ (E-commerce App XYZ)

**Sprint Goal:** Hoàn thành toàn bộ trải nghiệm Tìm kiếm sản phẩm (từ nhập liệu, hiển thị kết quả, đến lọc cơ bản).

### (A) Mục tiêu của buổi Sprint Review

Mục tiêu chính của buổi Sprint Review cho tính năng **Tìm kiếm sản phẩm** là:

- Xác nhận hoạt động đúng:** Xác minh rằng chức năng tìm kiếm hoạt động ổn định và chính xác trên môi trường Staging/Pre-Production (ví dụ: kết quả tìm kiếm phải trả về chính xác sản phẩm có từ khóa đó).
- Thu thập phản hồi về trải nghiệm người dùng (UX):** Lấy ý kiến từ Stakeholders về tốc độ tìm kiếm, tính trực quan của giao diện hiển thị kết quả, và sự tiện lợi của bộ lọc cơ bản (ví dụ: lọc theo giá, danh mục).
- Đánh giá giá trị:** Cùng nhau đánh giá liệu tính năng vừa hoàn thành có mang lại giá trị gia tăng như kỳ vọng cho người dùng và doanh nghiệp hay không.
- Ưu tiên cải tiến:** Dựa trên phản hồi, cập nhật Product Backlog và xác định các cải tiến cho tính năng Tìm kiếm (ví dụ: thêm tính năng gợi ý từ khóa, tìm kiếm nâng cao) để chuẩn bị cho Sprint tiếp theo.

### (B) Ai sẽ nói gì trong 1 phút mở đầu cuộc họp

Vai trò	Lời thoại (Mô phỏng nguyên văn)
<b>Product Owner (PO) - Lan</b>	"Chào mọi người, cảm ơn đã tham gia. Sprint Goal của chúng ta là ' <b>Hoàn thành toàn bộ trải nghiệm Tìm kiếm sản phẩm</b> '. Chúng tôi ưu tiên tính năng này vì dữ liệu cho thấy 40% người dùng rời bỏ ứng dụng khi không tìm thấy sản phẩm nhanh chóng. Hôm nay, chúng ta sẽ xem xét và cho ý kiến về luồng tìm kiếm và lọc sản phẩm. Sau buổi này, chúng tôi cần quyết định liệu tính năng đã sẵn sàng để triển khai rộng rãi hay chưa."
<b>Scrum Master (SM) - Minh</b>	"Vâng, xin lưu ý cuộc họp kéo dài 30 phút và chúng ta tập trung vào Product Increment (Tính năng đã hoàn thành). Sau demo, chúng ta sẽ dành 10 phút để phản hồi và thảo luận. Xin hãy giữ các câu hỏi về quy trình hoặc nhân sự cho buổi Retrospective sau nhé."
<b>Dev Lead - Hùng</b>	"Tôi là Hùng, phụ trách demo. Tôi sẽ trình bày luồng chính: tìm kiếm cơ bản và áp dụng bộ lọc. Tốc độ xử lý đã được tối ưu, và chúng tôi tin rằng trải nghiệm tìm kiếm đã được cải thiện đáng kể. Xin mời PO Lan bắt đầu với phần Demo."

### (C) Dev sẽ demo tính năng như thế nào

Dev Lead Hùng sẽ thực hiện demo theo các bước chi tiết sau:

- Bước mở giao diện và luồng chính:** Hùng mở ứng dụng Thương mại điện tử trên điện thoại/máy tính, điều hướng đến màn hình chính và click vào **thanh tìm kiếm (Search Bar)**.
- Dữ liệu mẫu và luồng thành công:** Hùng nhập từ khóa mẫu "laptop" và nhấn Enter. Màn hình ngay lập tức hiển thị **kết quả tìm kiếm** (danh sách các sản phẩm là laptop) với tốc độ tải dưới 1 giây.
- Demo tính năng phụ (Bộ lọc):** Sau đó, Hùng thực hiện **lọc kết quả** bằng cách click vào nút "Bộ lọc", chọn "**Giá từ 10.000.000 VNĐ đến 20.000.000 VNĐ**" và "**Danh mục: Laptop gaming**". Danh sách sản phẩm được cập nhật ngay lập tức chỉ hiển thị các sản phẩm thỏa mãn 2 tiêu chí lọc này.
- Tình huống lỗi (Boundary/Edge Case):** Hùng tiếp tục nhập từ khóa "abcxyz" (một từ khóa không có kết quả trong hệ thống). Màn hình hiển thị thông báo thân thiện "**Rất tiếc, không tìm thấy sản phẩm nào phù hợp**" kèm theo gợi ý các danh mục phổ biến, thay vì bị crash hoặc đứng máy.
- Kết quả mong đợi được hiển thị cho stakeholders:** Hùng nhấn mạnh hai điểm: (1) **Tốc độ phản hồi nhanh (< 1 giây)** khi tìm kiếm và lọc, và (2) **Giao diện thân thiện** với người dùng, đảm bảo ngay cả khi không có kết quả, người dùng vẫn được giữ lại trên ứng dụng.

## Bài 6:

Tên Retrospective: "**Hành Trình Tăng Tốc**" (Focusing on eliminating slow-downs and late deliveries)

### Kỹ thuật Sử dụng:

- 4Ls (Liked – Learned – Lacked – Longed for):** Sử dụng để thu thập ý kiến một cách rộng rãi và có cấu trúc, giúp khơi gợi các ý tưởng cải tiến (điều mà team đang thiếu).
- 5 Whys:** Sử dụng trong giai đoạn "Generate Insights" để đào sâu tìm **nguyên nhân gốc rễ** của các vấn đề quan trọng như Daily Scrum muộn và Tester báo bug trễ.

---

### (B) Agenda Chi tiết Từng Giai đoạn (90 Phút)

Giai đoạn	Hoạt động	Thời lượng	Mục tiêu

<b>1. Set the Stage</b>	<b>Check-in &amp; Nguyên tắc (5 phút):</b> Scrum Master nhắc lại mục đích (cải tiến liên tục), đảm bảo không gian an toàn (luật Chatham House Rule), và thực hiện một hoạt động khởi động nhanh (ví dụ: "Hãy nói một từ mô tả cảm xúc của bạn về Sprint vừa qua").	<b>5 phút</b>	Tạo không gian an toàn, tập trung năng lượng.
<b>2. Gather Data</b>	<b>Kỹ thuật 4Ls (Liked – Learned – Lacked – Longed for) (20 phút):</b> Mỗi thành viên viết các ghi chú dán (sticky notes) cho 4 mục: <i>Liked</i> (Thích), <i>Learned</i> (Học được), <i>Lacked</i> (Thiếu sót), và <i>Longed for</i> (Mong muốn). PO/SM sắp xếp các ghi chú theo nhóm, đặc biệt chú ý đến mục <i>Lacked</i> (nơi các vấn đề như bug muộn, Daily muộn sẽ xuất hiện).	<b>20 phút</b>	Thu thập dữ liệu toàn diện, xác định vấn đề nóng.
<b>3. Generate Insights</b>	<b>a. Thảo luận và Bỏ phiếu (15 phút):</b> Cả nhóm thảo luận nhanh về các điểm đã được nhóm lại. Sau đó, mỗi người dùng 3 điểm để bỏ phiếu cho 2-3 vấn đề quan trọng nhất cần giải quyết (ví dụ: Daily muộn và Báo bug trễ). <b>b. Phân tích gốc rễ (5 Whys) (20 phút):</b> Nhóm tập trung vào vấn đề có nhiều phiếu bầu nhất ("Daily Scrum muộn 10-15 phút") và áp dụng kỹ thuật 5 Whys để đào sâu, tìm ra nguyên nhân cốt lõi (ví dụ: Vì sao muộn? -> Vì mọi người chưa chuẩn bị -> Vì không rõ mục tiêu/chưa xem board trước ->...).	<b>35 phút</b>	Phân tích các vấn đề ưu tiên, tìm ra nguyên nhân gốc rễ.
<b>4. Decide What to Do</b>	<b>Lập Kế hoạch Hành động (20 phút):</b> Dựa trên nguyên nhân gốc rễ vừa tìm được, cả nhóm cùng đề xuất các giải pháp khả thi. Scrum Master viết 5 Action Items <b>SMART</b> cụ thể, giao trách nhiệm (Ai), và đặt thời hạn (Khi nào) ngay tại cuộc họp.	<b>20 phút</b>	Đưa ra các Action Items cụ thể và cam kết thực hiện.
<b>5. Close the Retro</b>	<b>Feedback và Cảm ơn (10 phút):</b> (a) Mời một thành viên xung phong làm "chủ sở hữu" theo dõi Action Items này cho đến buổi Retro tiếp theo. <b>(b) Hoạt động Check-out:</b> Đánh giá mức độ hiệu quả	<b>10 phút</b>	Kết thúc tích cực, đảm bảo cam kết theo dõi, và lấy

	của buổi Retro (ví dụ: thang điểm 1-5). <b>(c) Cảm ơn mọi người.</b>		feedback về chính buổi Retro.
<b>Tổng cộng</b>		<b>90 phút</b>	

**(C) 5 Action Items Cụ thể**

Ai	Làm gì (Action)	Khi nào (Deadline)	Đo lường thế nào (How to Measure)	Vấn đề giải quyết
<b>Scrum Master (SM) + Team</b>	Thử nghiệm đặt báo thức 5 phút trước giờ Daily Scrum (9:25 sáng) và đóng cửa phòng họp/cuộc gọi lúc 9:30.	Bắt đầu ngay Sprint tiếp theo	Theo dõi sự tuân thủ trong 10 ngày Daily đầu tiên. Mục tiêu: 10/10 lần Daily bắt đầu đúng giờ.	Daily Scrum muộn
<b>Tester (Minh) + Dev Lead (Tùng)</b>	Thiết lập quy trình: Mọi task đã "Done for Testing" (D4T) phải được kiểm thử trong vòng tối đa <b>4 giờ làm việc</b> kể từ khi nhận. Nếu quá thời gian, Dev Lead cần được thông báo ngay.	Bắt đầu ngay Sprint tiếp theo	Giảm thiểu số lượng bug được báo cáo trong 2 ngày cuối Sprint xuống <b>dưới 20%</b> tổng số bug.	Tester báo bug muộn
<b>Dev Lead (Tùng)</b>	Dành 30 phút vào giữa Sprint (Ngày 7) để hướng dẫn team cách thực hiện <b>Unit Test</b> hiệu quả hơn, tập trung vào các tình huống biên (edge cases) và lỗi nhập liệu phổ biến.	Trước Ngày 7 của Sprint tiếp theo	Giảm số lượng bug nghiêm trọng (Severity 1 & 2) phát sinh từ Dev Testing xuống <b>0</b> .	Bug dồn dập, task spillover
<b>Product Owner (PO) + Team</b>	Trước Daily Scrum, PO/SM sẽ dành 1 phút để hỏi <b>một câu hỏi cụ thể</b> về quy trình (ví dụ: "Có rào cản nào	Bắt đầu ngay Sprint tiếp theo	Số lượng ý kiến cải tiến được đề xuất trong 4Ls của Retro sau tăng	Ít ý kiến cải tiến

	ngăn bạn hoàn thành task ngày hôm qua không?"') thay vì chỉ hỏi "Hôm nay làm gì?".		<b>25% so với lần này.</b>	
<b>Cả Team</b>	Phân tích 3 task bị <i>spillover</i> gần đây nhất, xác định lại Definition of Done (DoD) và cam kết chỉ đưa vào Sprint Backlog những PBI mà <b>mọi người đều hiểu rõ</b> DoD của nó.	Ngay trong buổi Sprint Planning tiếp theo	Giảm số lượng task bị <i>spillover</i> trong Sprint tiếp theo xuống <b>tối đa 1 task</b> .	Task bị spillover

### Bài 7:

Format	Thời gian đề xuất	Phù hợp khi team...	Ưu điểm	Nhược điểm	Điểm số dễ dẫn dắt (1-5)
<b>Start Stop Continue (SSC)</b>	60 - 90 phút	Cần tập trung vào <b>hành động cụ thể</b> và cải thiện quy trình ngay lập tức (What to do next).	Đơn giản, tập trung vào hành động, dễ biến đổi thành Action Items.	Có thể bỏ qua nguyên nhân gốc rễ sâu xa (root-cause).	4
<b>Mad Sad Glad (MSG)</b>	60 - 90 phút	Đang có vấn đề về <b>tinh thần, cảm xúc tiêu cực, stress, hoặc thiếu giao tiếp</b> (cần giải tỏa cảm xúc).	Khuyến khích chia sẻ cảm xúc thật, giải tỏa căng thẳng, tìm ra vấn đề "ngầm" về con người.	Cảm xúc có thể lấn át, cần Scrum Master khéo léo để chuyển từ cảm xúc sang hành động.	3

<b>4L (Liked, Learned, Lacked, Longed for)</b>	75 - 105 phút	Cần một cái nhìn <b>toàn diện, có cấu trúc</b> về cả thành công, bài học và điểm thiếu sót (tổng kết đa chiều).	Cân bằng (cả tích cực và tiêu cực), giúp team nhìn lại cả quá trình học hỏi và thiếu sót.	Mục "Longed for" có thể trở thành danh sách yêu cầu không thực tế hoặc quá mơ hồ.	4
<b>Sailboat</b>	90 - 120 phút	Đang cảm thấy có quá nhiều <b>rào cản/rủi ro</b> cản trở mục tiêu và cần cái nhìn <b>tổng thể</b> về chiến lược/tầm nhìn (tương lai).	Trực quan, dễ hiểu (dùng hình ảnh ẩn dụ), nhìn được rủi ro (đá ngầm) và cơ hội (gió).	Dài hơn các format khác, ít tập trung vào quy trình làm việc hằng ngày.	3
<b>KALM (Keep, Add, More, Less)</b>	60 - 90 phút	Đã thực hiện Retro nhiều lần và cần tập trung vào việc <b>tinh chỉnh tần suất/mức độ</b> của các hoạt động hiện tại (process tuning).	Tập trung vào hành động "lượng" (More/Less), rất phù hợp để tinh chỉnh quy trình.	Không khuyến khích chia sẻ cảm xúc, có thể bỏ lỡ các vấn đề mới (New).	4

## Bài 8:

### Kịch Bản Xử Lý Xung Đột trong Sprint Retrospective

**Vai trò:** Scrum Master

### (A) Bạn sẽ nói gì đầu tiên để lấy lại không khí?

Ngay khi nhận thấy căng thẳng (Developer A và Tester B cãi nhau, PO chen ngang, người dùng điện thoại), Scrum Master (SM) cần can thiệp ngay lập tức, không để cuộc tranh cãi kéo dài.

#### Lời thoại của SM:

"Cảm ơn mọi người. Tôi hiểu rằng Sprint vừa qua rất khó khăn và mọi người đều đang cảm thấy áp lực. Việc có cảm xúc là hoàn toàn tự nhiên, nhưng mục tiêu của chúng ta hôm nay không phải là tìm ra ai sai hay ai kém. **Mục tiêu là tìm ra những vấn đề trong QUY TRÌNH làm việc của chúng ta để giúp team làm việc TỐT HƠN trong Sprint sau.** Hãy tạm gác lại những cảm xúc cá nhân. Chúng ta hãy hít một hơi sâu. Để đảm bảo buổi họp hiệu quả và an toàn, chúng ta sẽ bắt đầu lại với **Luật Chatham House** - Mọi điều chúng ta nói ở đây sẽ được giữ kín trong phòng này, và chúng ta cam kết chỉ tập trung vào **sự kiện và hệ thống**, không đổ lỗi cá nhân. Hãy bắt đầu bằng cách tập trung năng lượng vào một hoạt động giúp chúng ta nhìn lại chuỗi sự kiện."

### (B) Bạn dùng kỹ thuật nào để mọi người không đổ lỗi cá nhân?

Kỹ thuật được chọn: Timeline (Dòng thời gian) kết hợp với Silent Writing (Viết thầm).

#### Lý do:

- **Silent Writing** (cho giai đoạn Gather Data) buộc mọi người phải viết ra **sự kiện** thay vì nói ra ý **kiến/cảm xúc** ban đầu, từ đó ngăn chặn việc đổ lỗi bằng lời nói ngay từ đầu.
- **Timeline** giúp cả nhóm nhìn thấy chuỗi sự kiện đã xảy ra: **Ngày X Dev A** hoàn thành code, **Ngày Y Tester B** nhận được code, **Ngày Z Bug** được báo cáo. Điều này chuyển trọng tâm thảo luận từ "Dev code bẩn" hoặc "Tester làm muộn" sang "Khoảng thời gian chờ giữa X và Y quá dài" hoặc "Bug đó xuất hiện do thiếu [quy trình]". Kỹ thuật này trực tiếp giải quyết vấn đề **đổ lỗi cá nhân và bug muộn** bằng cách tập trung vào dữ liệu và mốc thời gian.

### (C) 3 câu nói "thần chú" bạn sẽ dùng khi có người đổ lỗi

Khi có thành viên bắt đầu đổ lỗi ("Tại A làm cái này...") hoặc đi chêch khỏi chủ đề ("Cái này không quan trọng..." - của PO), SM dùng các câu sau để chuyển hướng thảo luận:

- "Khoan đã, hãy tập trung vào hệ thống/quy trình."** (*Khi Dev/Tester đổ lỗi*): "Tôi hiểu bạn đang bức bối, nhưng thay vì nói 'Code bẩn', chúng ta hãy cùng phân tích xem quy trình **Review Code** hoặc **Unit Test** của chúng ta đã bỏ sót điều gì?"
- "Giá trị nào bạn muốn mang lại?"** (*Khi PO chen ngang*): "Cảm ơn ý kiến của PO [tên], nhưng hiện tại chúng ta đang ở giai đoạn tìm kiếm **cơ hội cải tiến quy trình**. Yêu cầu của bạn có thể được đưa vào Product Backlog, nhưng ngay lúc này, điều quan trọng nhất mà bạn thấy team cần cải thiện trong quy trình là gì?"
- "Hãy giúp tôi hiểu điều gì đã xảy ra."** (*Câu thần chú chuyển hướng*): "Thay vì nói 'Anh/em làm muộn', chúng ta hãy nhìn vào Timeline. Điều gì đã xảy ra vào **khoảnh khắc** [sự kiện] này? Hãy mô tả sự kiện khách quan đó, và chúng ta sẽ cùng nhau tìm ra rào cản."

#### (D) Action Item bạn sẽ đưa ra để giải quyết tận gốc vấn đề đổ lỗi

Vấn đề gốc rễ của việc đổ lỗi là sự thiếu rõ ràng về trách nhiệm và chất lượng công việc (Dev không hiểu "sạch" là gì, Tester không hiểu "hoàn thành" là gì).

Ai	Làm gì (Action)	Khi nào (Deadline)	Đo lường thế nào (How to Measure)	Vấn đề giải quyết
<b>Cả Team + Dev Lead (A)</b>	Tổ chức một buổi họp 60 phút để <b>Cải tiến Definition of Done (DoD)</b> , bổ sung các tiêu chí định lượng (Quantitative Criteria) rõ ràng cho khâu Dev (ví dụ: yêu cầu độ bao phủ Unit Test tối thiểu 70%, mọi Pull Request phải có ít nhất 1 người review) và cho khâu Testing (ví dụ: môi trường staging phải được test smoke).	Trong vòng 3 ngày làm việc tới	Theo dõi trong Sprint tiếp theo: Số lượng tranh cãi/dổ lỗi giữa Dev-Tester tại Daily Scrum giảm 80% (Quan sát của SM).	Đổ lỗi cá nhân, Code bẩn
<b>Scrum Master (SM)</b>	Thiết lập "Hòm Thư An Toàn" (Safety Box) ẩn danh để thành viên ghi lại cảm xúc/vấn đề (kể cả người dùng điện thoại) trước 24 giờ diễn ra Retro. SM tổng hợp và thảo luận mà không tiết lộ danh tính.	Bắt đầu ngay Sprint tiếp theo	Tỷ lệ thành viên tham gia ý kiến trong Retro tăng 50% so với lần này.	Thiếu chia sẻ thật

<b>Cả Team</b>	Áp dụng quy tắc: <b>Mọi người đặt điện thoại ở chế độ Im lặng/Máy bay</b> và để úp xuống trong suốt 15 phút Daily Scrum và toàn bộ buổi Retrospective. SM chịu trách nhiệm nhắc nhở ngay lập tức nếu vi phạm.	Bắt đầu ngay Sprint tiếp theo	Tỷ lệ tuân thủ quy tắc điện thoại đạt 100% trong tất cả các cuộc họp có Team tham gia trong Sprint.	Thiếu tập trung/T ôn trọng
<b>Product Owner (PO) + SM</b>	Dành 5 phút cuối mỗi Daily Scrum để PO nhắc lại <b>giá trị của các PBI đang làm</b> (Business Value) để đội ngũ hiểu rõ hơn về tầm quan trọng của công việc, từ đó giúp PO không cảm thấy cần phải can thiệp vào quy trình kỹ thuật.	Bắt đầu ngay Sprint tiếp theo	PO không ngắt lời thảo luận quy trình của Dev/Tester trong 3 Daily Scrum/Retro tiếp theo.	PO can thiệp quá mức