

1:

- B1:

- Mục tiêu của Sprint là quản lý công việc nhóm của các thành viên
- Liệt kê các task (tính năng) để hoàn thành trong Sprint:
 - Tạo – chỉnh sửa – xóa task.
 - Gán task cho thành viên.
 - Thêm trạng thái tiến độ

- B2:

Các thành viên trong team Dev sẽ demo về tính năng đó

- Khi tạo task, người dùng chọn tên thành viên từ danh sách.
- Task sẽ hiển thị trong bảng Kanban của từng người

- B3:

- Tính năng hoạt động ổn. Tuy nhiên phần lịch sử cập nhật chỉ hiển thị thời gian, chưa thấy tên người chỉnh sửa. Nên bổ sung để dễ theo dõi.
- B4: Kết thúc cuộc họp

2:

- PBI: Thêm lọc thống kê theo ngày

Là một trưởng phòng vận hành, tôi muốn có bộ lọc thống kê theo ngày hoặc trong khoảng 24 giờ, nên tôi có thể xem dữ liệu chi tiết và đánh giá chính xác hoạt động của người dùng, thay vì chỉ xem dạng tổng hợp theo tuần

- Acceptance Criteria cho PBI:

AC1 – Lọc theo ngày cụ thể

- Given người dùng đang mở trang thống kê
- When người dùng chọn một ngày cụ thể
- Then hệ thống hiển thị thống kê hoạt động đúng trong ngày đó, không gom vào tuần.

AC2 – Lọc theo 24 giờ gần nhất

- Given người dùng đang xem thống kê
- When người dùng chọn tùy chọn 24 giờ gần nhất
- Then báo cáo được cập nhật với dữ liệu chính xác trong 24 giờ tính từ thời điểm hiện tại.

AC3 – Không có dữ liệu

- Given người dùng chọn một khoảng thời gian không có thống kê
- When hệ thống xử lý bộ lọc
- Then hiển thị thông báo Không có dữ liệu và không hiển thị biểu đồ sai lệch.

3:

Ngày	Công việc
1	Sprint Planning
2	Cơ sở dữ liệu
3	API chức năng quản lý môn học
4	Bug API không upload được ảnh
5	Fix lỗi API không upload được ảnh
6	Xây dựng UI Quản lý môn học
7	Validate thêm sửa xóa môn học
8	Bug UI về vấn đề vỡ giao diện
9	Fix lỗi UI về vấn đề giao diện
10	Ghép nối UI với API
11	Test trải nghiệm người dùng
12	Gặp lỗi vấn đề lọc thông tin môn học hoặc tìm kiếm
13	Sửa lỗi vấn đề lọc và tìm kiếm

14	Review hoặc relase demo
----	-------------------------

Ngày	Sự kiện	Mô tả	Ảnh hưởng	Bài học
4	Bug API upload ảnh	Khi thử upload ảnh môn học, API báo lỗi	Tiêu cực – trì hoãn tích hợp UI	Test API sớm để phát hiện lỗi trước khi UI hoàn thiện
5	Fix lỗi API	Dev sửa lỗi upload ảnh thành công	Tích cực – API hoạt động đúng	Giữ checklist QA cho mỗi endpoint
8	Bug UI vỡ giao diện	Giao diện bị vỡ khi thêm môn học dài	Tiêu cực – ảnh hưởng trải nghiệm test	Luôn test responsive và edge case UI sớm
12	Xác định lỗi lọc/tìm kiếm	Phát hiện thuật toán filter chưa chính xác	Tiêu cực – ảnh hưởng trải nghiệm người dùng	Phân tích logic trước khi code; test dữ liệu thực
14	Review / Release demo	Demo hoàn thành Sprint cho PO và stakeholders	Tích cực – Sprint đạt mục tiêu	Duy trì quy trình release demo để nhận feedback sớm

4:

Trong bối cảnh team mệt mỏi, ít giao tiếp, stress cao, bug dồn cuối Sprint, kỹ thuật Retrospective phù hợp nhất là: **Mad – Sad – Glad.**

Lý do lựa chọn:

- Giải tỏa cảm xúc: Mad–Sad–Glad khuyến khích các thành viên chia sẻ cảm xúc tiêu cực (Mad, Sad) và tích cực (Glad) về Sprint, giúp giảm stress và “nút thắt” tâm lý.
- Khơi gợi sự chia sẻ thật: Việc phân loại cảm xúc giúp team cởi mở hơn, nói thẳng vấn đề mà không sợ bị đánh giá.
- Tìm lại tinh thần hợp tác: Khi mọi người thấy người khác cũng gặp khó khăn, tinh thần đồng đội được củng cố.
- Phân tích nguyên nhân gốc rễ: Sau khi chia sẻ cảm xúc, Scrum Master có thể dẫn dắt nhóm thảo luận về nguyên nhân stress, bug dồn cuối Sprint và đưa ra cải tiến.

Tại sao các kỹ thuật khác ít phù hợp hơn:

- Start–Stop–Continue: tốt cho cải tiến quy trình, nhưng không giúp giải tỏa cảm xúc hay khơi gợi chia sẻ.
- 4Ls: thiên về học hỏi và thiếu sót, không tập trung vào cảm xúc cá nhân.
- Sailboat: phân tích tổng thể Sprint, phù hợp cho chiến lược hơn là giải tỏa tâm lý.
- Timeline: chỉ ghi lại sự kiện, không khuyến khích chia sẻ cảm xúc tiêu cực.

5:

(A) Mục tiêu Sprint Review

- Kiểm tra tính năng **Tìm kiếm sản phẩm** có hoạt động đúng yêu cầu.
- Thu thập phản hồi về tốc độ tìm kiếm và cách hiển thị kết quả.
- Xác nhận giao diện thân thiện, dễ dùng.
- Ghi nhận đề xuất cải tiến để đưa vào Product Backlog.

(B) Nội dung 1 phút mở đầu

Product Owner:

“Trong Sprint này, mục tiêu chính là hoàn thiện tính năng **Tìm kiếm sản phẩm** để cải thiện trải nghiệm người dùng và hỗ trợ tăng tỉ lệ mua hàng. Hôm nay, đội dev sẽ demo bản chạy thử để chúng ta xem kết quả.”

Scrum Master:

“Buổi Sprint Review kéo dài khoảng 30 phút. Chúng ta tập trung vào việc xem Increment và đưa ra phản hồi. Đây không phải buổi kiểm thử chi tiết.”

Dev Lead:

“Chúng tôi sẽ demo tìm kiếm theo từ khóa, lọc theo danh mục và cách hệ thống xử lý khi không có kết quả.”

(C) Dev demo tính năng – mô tả ngắn gọn

1. Dev mở trang chủ và chỉ vào thanh tìm kiếm.
2. Nhập từ khóa “tai nghe bluetooth” và chạy tìm kiếm.
3. Hệ thống hiển thị danh sách sản phẩm kèm giá, hình và đánh giá.
4. Dev chọn lọc theo danh mục để thu hẹp kết quả.
5. Thủ nhập từ khóa không tồn tại để kiểm tra thông báo “Không tìm thấy”.
6. Dev cho thấy tốc độ phản hồi <1 giây như yêu cầu.
7. Cuối cùng click vào 1 sản phẩm để chứng minh luồng tìm kiếm → xem chi tiết hoạt động trọn tru.

6:

(A) Tên buổi Retrospective + Kỹ thuật

Retrospective: Improve the Flow

Kỹ thuật: Start – Stop – Continue

(B) Agenda chi tiết – 90 phút

Thời lượng	Hoạt động
5 phút	Scrum Master mở đầu, nhắc mục tiêu Retro, tạo không khí an toàn chia sẻ
10 phút	Check-in nhanh: mỗi người nói 1 câu Điều khó nhất trong Sprint vừa rồi.
15 phút	Thu thập ý kiến theo 3 nhóm: <i>Start – Stop – Continue</i>

20 phút	Nhóm và thảo luận: gom các note trùng nhau, xác định 3–4 vấn đề lớn
20 phút	Thảo luận sâu từng vấn đề và đề xuất giải pháp thực tế
15 phút	Chốt Action Items
5 phút	Kết thúc: Mỗi người nói 1 câu Một điều tôi cam kết cải thiện Sprint tới

(C) Action (ít nhất 5– rõ ràng, đo được

1. Daily Scrum on-time

- Ai: Scrum Master
- Làm gì: Thiết lập đồng hồ bấm giờ + nhắc cả team vào họp đúng 9:00
- Khi nào: Từ Sprint kế tiếp, áp dụng hằng ngày
- Đo lường: Daily muộn ≤ 1 lần/tuần

2. Kiểm soát Spillover Task

- Ai: Dev Lead
- Làm gì: Tạo quick checklist Definition of Ready cho task trước khi đưa vào Sprint
- Khi nào: Hoàn thành trong tuần đầu Sprint
- Đo lường: Số task spillover giảm ít nhất 50% trong Sprint tiếp theo

3. Test sớm hơn

- Ai: Tester + Dev Lead
- Làm gì: Chia nhỏ task để bàn giao bản build sớm (ít nhất 3 lần trong Sprint thay vì cuối Sprint)
- Khi nào: Bắt đầu áp dụng từ Sprint tới
- Đo lường: Bug critical không xuất hiện vào ngày cuối Sprint

4. Tăng tương tác cải tiến

- Ai: Scrum Master

- Làm gì: Thực hiện mini-retro 10 phút vào giữa Sprint để team đưa ý kiến sớm
- Khi nào: Sprint kế tiếp
- Đo lường: Số ý kiến cải tiến ≥ 5 trong buổi Retro cuối Sprint

5. Theo dõi bug & chất lượng sớm

- Ai: Tester
- Làm gì: Tạo bảng visible Bug trend trong Sprint
- Khi nào: Trong 2 ngày đầu Sprint
- Đo lường: Tester báo bug trong 3–5 ngày đầu Sprint, không dồn cuối Sprint

7:

Format	Thời gian đề xuất	Phù hợp khi team	Ưu điểm	Nhược điểm
Start – Stop – Continue	45–60 phút	Cần cải tiến nhanh, muốn tập trung vào hành vi và quy trình	Dễ hiểu, dễ nhóm ý, ra action items rõ ràng	Ít đào sâu cảm xúc, có thể bỏ sót nguyên nhân gốc rễ
Sailboat	60-90 phút	Team muốn nhìn tổng thể về mục tiêu, rủi ro và động lực	Tạo hình ảnh trực quan, khuyến khích phân tích sâu	Mất thời gian chuẩn bị, dễ bị lan man
Mad Sad Glad	45–60 phút	Team cần bộc lộ cảm xúc, làm rõ xung đột hoặc căng thẳng	Khoi gợi chia sẻ thật lòng, tốt cho xây dựng team	Có thể dẫn tới tranh luận cá nhân, thiếu hướng cải tiến cụ thể

4L (Liked, Learned, Lacked, Longed for)	60 phút	Team muốn đánh giá toàn diện cả học được, thiếu gì, mong muốn gì	Bao quát—vừa cảm xúc, vừa kiến thức, vừa thiếu sót	Nhiều mục nên mất thời gian chia sẻ
KALM (Keep, Add, More, Less)	45–60 phút	Đã hoạt động ổn, muốn tinh chỉnh tối ưu	Rất rõ ràng cho cải tiến liên tục, dễ tạo action items	Không phù hợp khi team có nhiều vấn đề lớn cần giải quyết

8:

(A) Mở đầu lấy lại không khí

- “Mục tiêu Retro là chia sẻ Sprint và cải tiến quy trình, không tìm người chịu lỗi.”
- “Hãy nói về việc xảy ra, không nói về cá nhân.”
- “Nếu nóng, hãy viết ra sticky note, chúng ta sẽ thảo luận sau.”

(B) Kỹ thuật tránh đồ lỗi

- Silent Writing + Mad/Sad/Glad: mỗi người ghi ý kiến riêng về Sprint (Mad/Sad/Glad), tập trung vào sự kiện, không nhắm vào cá nhân.
- Gom nhóm theo chủ đề, thảo luận nguyên nhân và giải pháp.

(C) 3 câu “thần chú”

- “Hãy nói về việc, không nói về người.”
- “Cùng học từ vấn đề, không tìm kẻ sai.”
- “Mục tiêu là cải tiến Sprint, không phải phê phán.”

(D) Action Items

- Scrum Master + Team: Thiết lập Definition of Done & Acceptance Criteria → giảm bug/spillover.

- Dev & Tester: Daily sync 5 phút → tránh delay task.
- Team: Code review & test review định kỳ → nâng chất lượng code.
- Scrum Master: Duy trì quy tắc Retro → tập trung vào việc, không đổ lỗi