

Bài 1:

1. Kế hoạch tổ chức Daily Scrum

Mục tiêu của buổi họp

- Nắm bắt tình hình công việc hàng ngày của từng thành viên.
- Phát hiện sớm các điểm nghẽn có thể ảnh hưởng đến Sprint Goal.
- Đảm bảo cuộc họp diễn ra ngắn gọn, không vượt quá 15 phút.

Cách triển khai

- Cuộc họp được tổ chức đúng giờ mỗi ngày và ở một địa điểm cố định để tạo thói quen.
- Hình thức họp đúng nhằm tăng sự tập trung và giảm thời gian lan man.
- Scrum Master chỉ dẫn dắt, còn việc cập nhật tiến độ là trách nhiệm của các Developer.
- Mỗi thành viên báo cáo dựa trên 3 câu hỏi tiêu chuẩn:
 1. Hôm qua tôi đã hoàn thành những gì để hướng tới Sprint Goal?
 2. Hôm nay tôi dự định làm gì?
 3. Có khó khăn nào đang cản trở tôi không?

2. Ví dụ buổi Daily Scrum

Scrum Master:

“Chúng ta bắt đầu Daily. Mỗi người báo cáo ngắn gọn theo 3 câu hỏi, mọi trao đổi chi tiết sẽ chuyển sang cuộc họp sau.”

Thành viên A

- Hôm qua: Hoàn thiện layout trang đăng nhập.
- Hôm nay: Tích hợp API đăng nhập.
- Trở ngại: API trả về status lỗi 500.

Scrum Master:

“Ghi nhận. A trao đổi với backend sau buổi họp, tôi sẽ tham gia hỗ trợ.”

Thành viên B

- Hôm qua: Đã làm xong 70% phần hiển thị danh sách sản phẩm.
- Hôm nay: Bổ sung chức năng lọc sản phẩm theo danh mục.
- Trở ngại: Thiếu dữ liệu test phù hợp.

Scrum Master:

“Tôi sẽ xin PO cung cấp bộ dữ liệu test trước 11 giờ.”

Thành viên C

- **Hôm qua:** Hoàn thiện test case cho module giỏ hàng.
- **Hôm nay:** Thực hiện kiểm thử tự động.
- **Trở ngại:** Không có.

Scrum Master kết thúc:

“Tốt lắm! Sau họp, A làm việc với backend về issue API. Tôi sẽ gửi dữ liệu cho B. Daily kết thúc.”

3. Xử lý các vấn đề thường gặp khi họp Daily

Tình huống phát sinh	Cách Scrum Master xử lý
Thành viên gấp lối kỹ thuật	Hẹn buổi trao đổi ngắn ngay sau Daily, chỉ những người liên quan tham gia.
Một người nói quá dài	Nhắc nhở: “Phần này chúng ta trao đổi sau nhé để giữ đúng thời lượng Daily.”
Cuộc họp chuyển sang tranh luận kỹ thuật	Ghi chú lại và mở buổi thảo luận riêng sau Daily.
Thành viên không chuẩn bị	Yêu cầu cập nhật task trên board từ hôm trước hoặc kiểm tra nhanh trước giờ họp.

4. Tình huống thực tế & cách giải quyết

◆ **Tình huống 1: Thành viên hoàn thành chậm**

- **Nguyên nhân:** Task phức tạp, Dev A không kịp tiến độ.
 - **Hướng xử lý:**
 - Tổ chức pair-programming ngay sau Daily với một Dev có kinh nghiệm.
 - Xem xét chia task lớn thành nhiều phần nhỏ hơn để dễ hoàn thành.
-

◆ **Tình huống 2: Công việc bị “block” do chờ người khác**

- **Nguyên nhân:** Dev B phải đợi backend cung cấp API mới.
- **Cách giải quyết:**
 - Scrum Master nhắc backend hoàn thiện sớm.
 - Cho phép Dev B dùng mock API để không bị đứng hình.

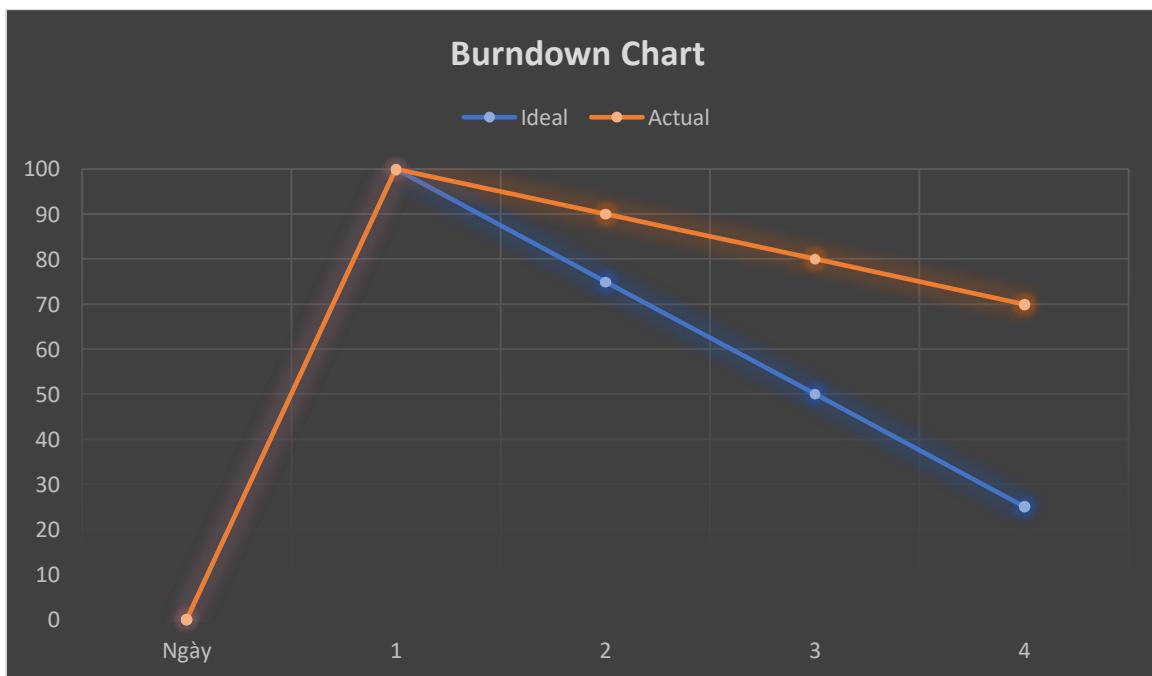
◆ Tình huống 3: PO chưa cung cấp tài liệu

- **Nguyên nhân:** Tester không có tài liệu chi tiết để viết test case.
 - **Cách xử lý:**
 - Scrum Master làm việc trực tiếp với PO để bổ sung tài liệu trong ngày.
 - Tester dựa trên User Story và tiêu chí chấp nhận (AC) để viết test tạm trước.
-

◆ Tình huống 4: Thành viên không chia sẻ trớ ngại

- **Cách giải quyết:**
 - Nhắc nhở nhẹ: “Daily là dịp để chúng ta hỗ trợ nhau, vấn đề nào cũng nên nói.”
 - Sau họp, Scrum Master trao đổi riêng để nắm rõ lý do và giúp mở lòng hơn trong các buổi tiếp theo.

Bài 2:



BÀI 2 – Burndown Chart

1. Burndown Chart Sprint – Mô tả tổng quan

Burndown Chart là biểu đồ thể hiện tiến độ “đốt điểm” (Story Points) theo từng ngày trong Sprint.

- Trục X: các ngày trong Sprint.
- Trục Y: tổng số Story Points còn lại chưa hoàn thành.

Ví dụ Sprint có 100 điểm, nhóm theo dõi tiến độ trong 5 ngày đầu.

2. Mô phỏng tiến độ 5 ngày đầu

Ngày	SP còn lại	Công việc ghi nhận
Ngày 1	100	Sprint bắt đầu – chưa hoàn tất task nào
Ngày 2	88	Hoàn thành 12 SP
Ngày 3	78	Hoàn thành 10 SP
Ngày 4	68	Hoàn thành 10 SP
Ngày 5	55	Hoàn thành 13 SP

3. Nhóm có đang đúng hướng không?

Nhìn vào chuỗi giảm:

$100 \rightarrow 88 \rightarrow 78 \rightarrow 68 \rightarrow 55$

Đây là biểu hiện của một Sprint “khỏe”, vì:

- Công việc được chia nhỏ hợp lý và được “kéo về Done” đều mỗi ngày.
 - Không xuất hiện điểm nghẽn rõ ràng.
 - Khả năng Sprint Goal hoàn thành là rất cao.
-

4. Trường hợp nhóm đi lệch tiến độ

Ví dụ đường thực tế như sau:

- Ngày 1: 100
- Ngày 2: 97
- Ngày 3: 95
- Ngày 4: 92
- Ngày 5: 80

=> Đến ngày thứ 5 chỉ giảm được 20 SP, trong khi kế hoạch cần khoảng 45 SP.

Đường biểu đồ sẽ nằm trên đường lý tưởng → dấu hiệu hụt hơi, dễ fail Sprint.

5. Scrum Master cần làm gì khi tiến độ chậm?

Bước 1 – Đánh giá nguyên nhân

Một số lý do phổ biến:

- User Story quá lớn hoặc chưa được tách nhỏ.
- Thành viên gặp lỗi kỹ thuật nhưng chưa nêu ra trong Daily.
- Yêu cầu thay đổi đột ngột từ PO.
- Cách phối hợp giữa frontend – backend – tester chưa hiệu quả.

Bước 2 – Họp Impediment Meeting (10–15 phút)

Đây không phải Daily, mà là buổi giải quyết vấn đề:

- Ai đang bị kẹt?
- Cần sự hỗ trợ từ ai?
- Có thể dùng mock data hoặc giải pháp tạm thời không?

Bước 3 – Sắp xếp lại thứ tự công việc

- Đẩy những User Story quan trọng nhất lên trước.
- Những việc không quan trọng hoặc không ảnh hưởng Sprint Goal có thể lùi lại (PO phải đồng ý).

Bước 4 – Tăng tốc kỹ thuật

- Pair programming với task khó.
- Mời chuyên gia trong team hỗ trợ nhanh.
- Ưu tiên hoàn thành từng task nhỏ hơn để kéo điểm về Done sớm.

Bước 5 – Cập nhật biểu đồ sau khi điều chỉnh

Nhóm cập nhật lại số điểm thực tế để theo dõi xem tiến độ đã được cải thiện hay chưa.

6. Khi Burndown Chart không giảm đều – Phân tích & hướng xử lý

Tình huống:

Đến ngày 5 còn 80 SP, trong khi lý tưởng phải còn khoảng 60 SP.

Nguyên nhân có thể:

- Story ước lượng sai hoặc quá “to”.
- Phát sinh nhiều task phụ ngoài kế hoạch.
- Tester chưa có dữ liệu kiểm thử.
- Backend chưa cung cấp API nên frontend bị đứng task.
- Dev làm xong nhưng chưa kéo sang “Done” vì thiếu DoD (Definition of Done).

Giải pháp đề xuất:

- Kiểm tra lại DoD để thống nhất “như thế nào là Done”.
- Chia nhỏ Story để tránh việc đến cuối Sprint mới hoàn thành hàng loạt.
- Tăng giao tiếp giữa dev và tester để hạn chế chờ nhau.
- Scrum Master làm việc với PO để loại bỏ yêu cầu không thật sự cần trong Sprint.
- Điều chỉnh Velocity và phân lại công việc phù hợp trong các ngày còn lại.

Bài 3:

Daily Scrum – Ngày thứ 6

1. Mô phỏng cuộc họp – Ngày thứ 6 của Sprint

Nhóm bước vào ngày thứ 6 với mục tiêu rà soát tiến độ và nhận diện các trở ngại.

Mỗi thành viên lần lượt báo cáo theo 3 câu hỏi quen thuộc, và Scrum Master ghi nhận để xử lý sau cuộc họp.

2. Các vấn đề được đưa ra & cách nhóm xử lý

◆ Vấn đề 1 – Dev A thiếu tài liệu chi tiết về API

Tình trạng: Không có mô tả cấu trúc response → khó tích hợp.

Hướng giải quyết:

- Dev B (backend) sẽ bổ sung tài liệu response sau khi Daily kết thúc.
 - Scrum Master đề nghị Dev B cập nhật đầy đủ vào Confluence trước cuối ngày để tránh tái diễn.
-

◆ Vấn đề 2 – Dev C không rõ ý nghĩa trường “role”

Tình trạng: Chưa hiểu mục đích trường “role” khi xử lý phân quyền.

Hướng giải quyết:

- Dev B xác nhận “role” dùng để định nghĩa quyền người dùng.
 - Hai bên thống nhất họp nhanh khoảng 10 phút sau Daily để làm rõ chi tiết.
-

◆ Vấn đề 3 – Giao diện Figma chưa có guideline mới

Tình trạng: Dev D không đủ guideline để tiếp tục coding.

Hướng giải quyết:

- Scrum Master nhắn designer cập nhật guideline vào buổi trưa.
- Trong lúc chờ, Dev D tạm sử dụng template tiêu chuẩn của team để không chậm tiến độ.

⚠ Lưu ý:

Những phần phân tích – trao đổi sâu chỉ được thực hiện sau Daily Scrum.
Trong buổi họp, team chỉ nên trỏ ngại, không bàn sâu vào kỹ thuật.

3. Bí quyết giữ Daily Scrum ngắn gọn và đúng mục đích

Thành viên tham gia: 4 Dev (Dev A, Dev B, Dev C, Dev D) + Scrum Master.

Dev A

- **Hôm qua:** Hoàn thành UI layout cho màn hình Login.
- **Hôm nay:** Kết nối API login và xử lý lỗi trả về.
- **Vấn đề:** Chưa nhận được tài liệu response format từ back-end → khó map dữ liệu.

Dev B

- **Hôm qua:** Tạo API đăng nhập, cấu hình JWT.
- **Hôm nay:** Viết API refresh token + unit test.
- **Vấn đề:** Không có gì cản trở.

Dev C

- **Hôm qua:** Thiết kế database cho module User.
- **Hôm nay:** Tạo migration và seed data.
- **Vấn đề:** Đang đợi Dev B xác nhận thêm trường “role” trong bảng user.

Dev D

- **Hôm qua:** Code phần xử lý validation phía client.
- **Hôm nay:** Tích hợp validation vào form đăng ký.
- **Vấn đề:** Figma chưa cập nhật guideline cho error message.

Daily Scrum – Ngày thứ 6

1. Mô phỏng cuộc họp – Ngày thứ 6 của Sprint

Nhóm bước vào ngày thứ 6 với mục tiêu rà soát tiến độ và nhận diện các rủi ro. Mỗi thành viên lần lượt báo cáo theo 3 câu hỏi quen thuộc, và Scrum Master ghi nhận để xử lý sau cuộc họp.

2. Các vấn đề được đưa ra & cách nhóm xử lý

❖ **Vấn đề 1 – Dev A thiếu tài liệu chi tiết về API**

Tình trạng: Không có mô tả cấu trúc response → khó tích hợp.

Hướng giải quyết:

- Dev B (backend) sẽ bổ sung tài liệu response sau khi Daily kết thúc.

- Scrum Master đề nghị Dev B cập nhật đầy đủ vào Confluence trước cuối ngày để tránh tái diễn.
-

❖ **Vấn đề 2 – Dev C không rõ ý nghĩa trường “role”**

Tình trạng: Chưa hiểu mục đích trường “role” khi xử lý phân quyền.

Hướng giải quyết:

- Dev B xác nhận “role” dùng để định nghĩa quyền người dùng.
 - Hai bên thống nhất họp nhanh khoảng 10 phút sau Daily để làm rõ chi tiết.
-

❖ **Vấn đề 3 – Giao diện Figma chưa có guideline mới**

Tình trạng: Dev D không đủ guideline để tiếp tục coding.

Hướng giải quyết:

- Scrum Master nhắn designer cập nhật guideline vào buổi trưa.
- Trong lúc chờ, Dev D tạm sử dụng template tiêu chuẩn của team để không chậm tiến độ.

⚠ **Lưu ý:**

Những phần phân tích – trao đổi sâu chỉ được thực hiện **sau** Daily Scrum.
Trong buổi họp, team chỉ nên trỏ ngại, không bàn sâu vào kỹ thuật.

3. Bí quyết giữ Daily Scrum ngắn gọn và đúng mục đích

Nguyên tắc	Giải thích
Chỉ 15 phút	Không kéo dài; Daily không phải sprint meeting thứ hai.
Báo cáo đúng 3 câu hỏi	Không mô tả kỹ thuật hoặc giải pháp trong Daily.
Hợp đứng	Tạo sự tập trung, tránh việc cuộc họp bị kéo dài.
Không xử lý vấn đề trong Daily	Chỉ ghi nhận, không phân tích. Việc xử lý diễn ra sau đó.
Scrum Master điều phối chặt chẽ	Nhắc khi thành viên lan man hoặc chuyển chủ đề.
Ghi chú impediments	Scrum Master note lại đầy đủ để follow-up.
Theo dõi Sprint Goal	Tập trung vào mục tiêu Sprint thay vì từng task nhỏ.

4. Vai trò của Scrum Master trong cuộc họp

1) Điều phối và giữ nhịp cuộc họp

- **Bắt đầu buổi Daily đúng thời điểm.**
 - **Hướng dẫn từng người trình bày theo 3 câu hỏi tiêu chuẩn.**
 - **Nhắc nhẹ khi cuộc họp có nguy cơ đi lệch hướng.**
-

2) Quản lý và theo dõi impediments

- **Ghi nhận vấn đề Dev A, C, D đưa ra.**
 - **Kết nối đúng người để giải quyết trong các buổi mini meeting sau đó.**
-

3) Bảo đảm tính minh bạch trong Sprint

- **Kiểm tra Sprint Board, đảm bảo task được cập nhật đúng trạng thái.**
 - **Nhắc thành viên kéo task đúng luồng: To Do → In Progress → Done.**
-

4) Ngăn Daily trở thành buổi họp kỹ thuật

Khi ai đó bắt đầu đi sâu vào chi tiết lỗi hoặc giải pháp, Scrum Master phải can thiệp ngay:

“Chúng ta tạm dừng phần này. Chuyển qua xử lý sau Daily, giờ mình tiếp tục cập nhật tiến độ.”

5) Tạo không khí cởi mở

- **Khuyến khích mọi người chia sẻ khó khăn.**
- **Hỗ trợ những thành viên còn ngại nêu vấn đề trước nhóm.**

Bài 4:



BUỚC 1 — Tạo Burndown Chart cho Sprint 10 ngày (120 story points)

Giả sử Sprint kéo dài **10 ngày làm việc**, tổng Story Points: **120**.

Ngày	Ideal Remaining (giảm đều)	Actual Remaining (thực tế)
Day 0	120	120
Day 1	108	112
Day 2	96	106
Day 3	84	100
Day 4	72	92
Day 5	60	85
Day 6	48	85 (<i>không giảm vì gấp ván đè</i>)
Day 7	36	70
Day 8	24	50
Day 9	12	20

Day 10	0	0 (<i>nhóm tăng tốc để hoàn thành Sprint</i>)
--------	---	---

BUỚC 2 — Mô phỏng cập nhật sau 5 ngày

Số story hoàn thành mỗi ngày

Ngày	Story hoàn thành	Story còn lại
Day 1	8	112
Day 2	6	106
Day 3	6	100
Day 4	8	92
Day 5	7	85

Nhận xét sau 5 ngày

- **Ideal còn lại** là: 60
- **Actual còn lại**: 85

=> Nhóm **đang chậm 25 điểm** so với tiến độ lý tưởng.

Biểu đồ (mô tả bằng lời)

- Đường **ideal**: giảm đều từ 120 → 0.
- Đường **actual**: giảm chậm hơn, luôn nằm **cao hơn** đường ideal.
- Tại ngày 5, đường actual bắt đầu **tách xa ideal**, báo hiệu nguy cơ không hoàn thành Sprint.

BUỚC 3 — Ngày thứ 6: Nhóm gấp ván đè → Burndown Chart thay đổi thế nào?

Tình huống ngày 6:

- Developer chính nghỉ đột xuất (thiếu tài nguyên).
- Một backend task bị block do lỗi môi trường.
- Không Story nào được hoàn thành trong ngày.

=> Story Points còn lại **giữ nguyên**: 85

Biểu đồ thể hiện điều gì?

- Đường actual tạo **một đoạn nằm ngang** vào ngày 6.
- Khoảng cách với đường ideal tiếp tục **mở rộng**.

Cách đội điều chỉnh:

1. **Scrum Master tổ chức họp nhanh** để xác định impediments.
2. **PO và team thống nhất re-prioritize backlog** trong Sprint.
3. **Pair programming** cho vài task quan trọng để tăng tốc.
4. **Chuyển 1 dev front-end sang hỗ trợ backend** tạm thời.
5. **Fix môi trường trong ngày**, gỡ block.

=> Từ ngày 7 trở đi team bắt đầu “đua tốc độ”.

Ngày	Story hoàn thành	Story còn lại
Day 7	15	70
Day 8	20	50
Day 9	30	20
Day 10	20	0

=> Nhóm hoàn thành Sprint đúng hạn.

BUỚC 4 — Vai trò của Scrum Master khi sử dụng Burndown Chart

Scrum Master có 3 nhiệm vụ lớn khi theo dõi Burndown Chart:

1, Phát hiện sớm các vấn đề (Early Detection)

Scrum Master cần nhận ra các dấu hiệu nguy hiểm:

Dấu hiệu	Nguyên nhân
Đường actual đi ngang	Team bị block, thiếu tài nguyên
Đường actual quá cao so với ideal	Ước lượng sai, làm việc không hiệu quả
Đường giảm “giật cục”	Công việc không được chia nhỏ đúng cách
Không giảm gì sau 1–2 ngày	Team có impedance nghiêm trọng

2, Hành động khi thấy tiến độ không đạt

Scrum Master cần:

- Loại bỏ impediments ngay lập tức.
- Tổ chức hậu Daily để giải quyết vấn đề.
- Nhắc team cập nhật board đúng trạng thái.
- Gợi ý chia task nhỏ hơn để kéo Story về Done nhanh hơn.
- Làm việc với Product Owner để **re-prioritize** task quan trọng.
- Bảo vệ team trước những gián đoạn từ bên ngoài.

3, Hướng dẫn team cải tiến

- Khuyến khích pair programming khi quá tải.
- Hỗ trợ team chạy kiểm thử sớm hơn để bắt lỗi nhanh.
- Đảm bảo task trong Sprint Backlog là khả thi.
- Dạy team cách ước lượng tốt hơn cho Sprint sau.

Bài 5:

Bước 1: Xây dựng Kanban Board cơ bản

Cấu trúc cột:

1. **To Do** – Công việc chưa bắt đầu.
2. **In Progress** – Công việc đang được thực hiện.
3. **Done** – Công việc đã hoàn thành.

Ví dụ thẻ công việc (Cards) cho Sprint:

Task	Mô tả
Thiết kế giao diện	Thiết kế UI/UX cho màn hình Dashboard và màn hình quản lý công việc
Phát triển API	Xây dựng API để lấy dữ liệu công việc và tạo công việc mới

Kiểm tra bảo mật	Kiểm thử lỗ hổng bảo mật cho API và giao diện người dùng
Tích hợp thanh toán	Tích hợp công thanh toán và xử lý đơn hàng
Viết unit test	Viết test cho các chức năng API đã phát triển
Cập nhật tài liệu	Viết hướng dẫn sử dụng và tài liệu kỹ thuật

Ghi chú: Mỗi thẻ công việc nên có thông tin về **người phụ trách, ngày bắt đầu, deadline, và ưu tiên**.

Bước 2: Đặt giới hạn công việc (Work In Progress – WIP Limit)

Ví dụ:

Cột	Giới hạn WIP
To Do	Không giới hạn
In Progress	3 nhiệm vụ
Done	Không giới hạn

Lý do đặt giới hạn WIP:

- Ngăn nhóm làm quá nhiều việc cùng lúc, tránh phân tán nỗ lực.
- Giúp nhóm tập trung hoàn thành nhiệm vụ đang làm trước khi bắt đầu nhiệm vụ mới.
- Giúp Scrum Master và nhóm dễ dàng nhận diện **tắc nghẽn (bottleneck)** trong quy trình.
- Khuyến khích cộng tác, hỗ trợ lẫn nhau để hoàn thành công việc.

Bước 3: Mô phỏng và xử lý tình huống quá tải

Tình huống:

- Cột **In Progress** có 6 nhiệm vụ, vượt quá giới hạn WIP là 3.
- Một số nhiệm vụ đang bị chậm, nhóm không biết ưu tiên xử lý việc nào trước.

Giải pháp tối ưu Kanban Board:

1. **Điều chỉnh WIP:** Giữ cột “In Progress” ở mức 3 nhiệm vụ, các nhiệm vụ còn lại phải đợi.
2. **Đánh dấu thẻ ưu tiên:** Sử dụng nhãn màu (Priority: High, Medium, Low) để nhóm biết việc nào cần xử lý trước.
3. **Thảo luận trong Daily Standup:** Nhận diện thẻ bị tắc nghẽn và hỗ trợ lẫn nhau.
4. **Tách nhiệm vụ lớn thành task nhỏ hơn:** Giúp hoàn thành nhanh và dễ di chuyển sang Done.
5. **Sử dụng cột bổ sung nếu cần:** Ví dụ cột “Blocked” cho những công việc gặp trở ngại bên ngoài.

Kết quả: Nhóm nhìn thấy ngay công việc đang bị tắc nghẽn, tập trung giải quyết và duy trì tiến độ.

Bước 4: Vai trò của Scrum Master với Kanban Board

- **Hướng dẫn nhóm sử dụng Kanban Board hiệu quả:** Giải thích cách di chuyển thẻ, cập nhật trạng thái, gắn nhãn ưu tiên.
- **Giám sát WIP Limit và tiến độ công việc:** Phát hiện tắc nghẽn, khuyến nghị điều chỉnh task hoặc hỗ trợ team.
- **Thúc đẩy hợp tác:** Khuyến khích team hỗ trợ nhau để hoàn thành công việc.
- **Đảm bảo minh bạch và cập nhật thường xuyên:** Board luôn phản ánh đúng trạng thái thực tế.
- **Cải tiến liên tục:** Quan sát dữ liệu từ Kanban Board để đưa ra các cải tiến quy trình, ví dụ như thay đổi cột, giới hạn WIP, hoặc tối ưu hóa task breakdown.

Bài 6:

1. Kế hoạch Daily Scrum

Cách tổ chức Daily Scrum chuẩn:

- Thời gian: 15 phút mỗi ngày, cố định vào cùng một giờ.
- Hình thức: đứng (stand-up) để hạn chế thời gian dài.
- Địa điểm: gần Kanban Board hoặc qua công cụ trực tuyến nếu nhóm remote.
- Mục tiêu: mọi thành viên cập nhật tình hình công việc, nhận diện vấn đề, đồng bộ thông tin.

Ba câu hỏi chính trong Daily Scrum:

Mỗi thành viên trả lời:

1. Hôm qua tôi đã hoàn thành những gì?
2. Hôm nay tôi dự định làm gì?
3. Có trở ngại nào cản trở công việc của tôi không?

Yếu tố quan trọng để Daily Scrum hiệu quả:

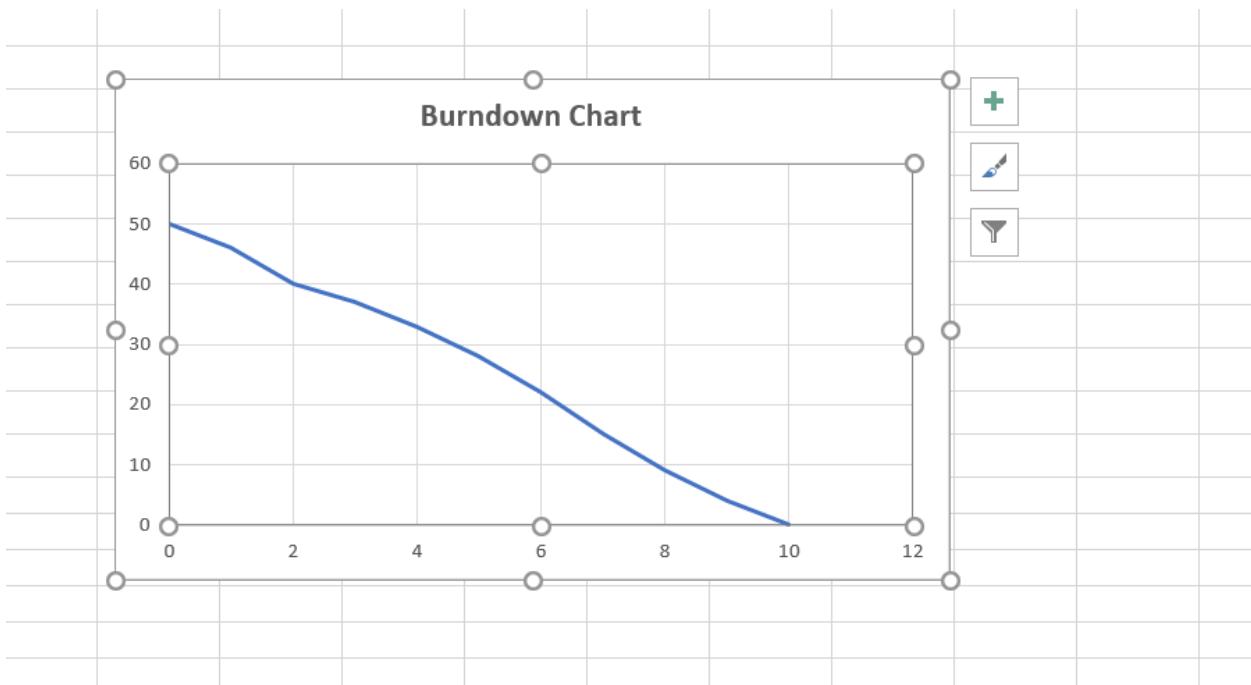
- Ngắn gọn, đúng trọng tâm (15 phút).
- Tập trung vào tiến trình và vấn đề, không thành báo cáo dài.
- Ghi nhận trở ngại và lập kế hoạch giải quyết sau cuộc họp.
- Cập nhật ngay Kanban Board hoặc công cụ quản lý công việc.

2. Burndown Chart

Thông tin Sprint:

- 10 User Stories, điểm Story từ 3–8, tổng 50 điểm.
- Thời gian Sprint: 10 ngày làm việc.

Ví dụ mô phỏng tiến độ 10 ngày đầu:



Giải thích đọc Burndown Chart:

- Trục X: ngày trong Sprint.
- Trục Y: số Story Points còn lại.
- Đường lý tưởng (Ideal Line): giảm đều từ 50 xuống 0.
- Đường thực tế: thể hiện tiến độ thực tế.
- **Phân tích:** nếu đường thực tế nằm trên đường lý tưởng → nhóm chậm tiến độ; nếu dưới → nhóm tiến nhanh.

3. Quản lý công việc bằng Kanban Board

Cột Kanban Board:

- **To Do:** các công việc chưa bắt đầu.
- **In Progress:** đang thực hiện.
- **Code Review:** hoàn tất code, đang review.
- **Done:** hoàn thành.

Ví dụ phân bổ công việc (từ 10 User Stories):

To Do	In Progress	Code Review	Done
Story 1 – Task A	Story 2 – Task B	Story 3 – Task C	Story 4 – Task D
Story 5 – Task E	Story 6 – Task F	Story 7 – Task G	Story 8 – Task H
Story 9 – Task I	Story 10 – Task J		

Sử dụng Kanban Board hiệu quả:

- Di chuyển công việc từ cột này sang cột khác khi tiến trình thay đổi.
- Nhóm dễ nhận biết công việc đang tắc nghẽn (bottleneck).
- Giúp Scrum Master và Product Owner theo dõi trực quan tiến độ.

4. Vai trò Scrum Master

Đảm bảo Daily Scrum hiệu quả:

- Nhắc nhở các thành viên tập trung vào 3 câu hỏi.
- Giữ thời gian ngắn gọn, tránh lan man.
- Cập nhật Kanban Board và giải quyết các vấn đề tồn đọng.

Giải quyết vấn đề và xung đột:

- Ghi nhận và tìm cách giải quyết trở ngại (blockers) ngay lập tức.
- Thúc đẩy trao đổi trực tiếp giữa các thành viên khi có xung đột.
- Hỗ trợ nhóm ưu tiên công việc theo mục tiêu Sprint.

Ba kỹ năng quan trọng của Scrum Master:

1. **Facilitation:** điều phối cuộc họp, giúp nhóm ra quyết định nhanh.
2. **Communication:** giao tiếp hiệu quả, truyền đạt rõ ràng thông tin.
3. **Conflict Resolution & Coaching:** giải quyết xung đột và hướng dẫn nhóm thực hành Scrum tốt.

Bài 7:

Bước 1: Kanban Board (To Do – In Progress – Done)

To Do	In Progress	Done
Thiết kế giao diện form đăng ký(3đ)	Kết nối API đăng ký (3đ)	Kiểm thử form đăng ký (3đ)

Bước 2: Tổng story points của Sprint

-Tổng: 9 Story Points

-Mỗi task: 3 điểm

+, Thiết kế giao diện form đăng ký → 3 điểm

+ , Kết nối API đăng ký → 3 điểm

+ , Kiểm thử form đăng ký → 3 điểm

Bước 3: Mô phỏng Daily Scrum buổi sáng

Thành viên 1 – Dev A

- , Hôm qua: Không có, vì Sprint bắt đầu hôm nay.

- , Hôm nay: Thiết kế giao diện form đăng ký.

=> Trở ngại: Chưa rõ yêu cầu chi tiết về trường dữ liệu từ PO.

Thành viên 2 – Dev B

- , Hôm qua: Không có.

- , Hôm nay: Kết nối API đăng ký, chuẩn bị cấu trúc request.

=> Trở ngại: Chờ giao diện từ Dev A để test API.

Thành viên 3 – Tester C

- , Hôm qua: Không có.

- , Hôm nay: Viết test case và chuẩn bị môi trường test.

- , Trở ngại: Chưa có bản build để kiểm thử form.

Bước 4: Cập nhật Burndown Chart (Cuối ngày)

Tiến độ thực tế:

- , Thiết kế UI → Hoàn thành (3đ)

- , Kết nối API → Hoàn thành (3đ)

- , Kiểm thử form → Hoàn thành (3đ)



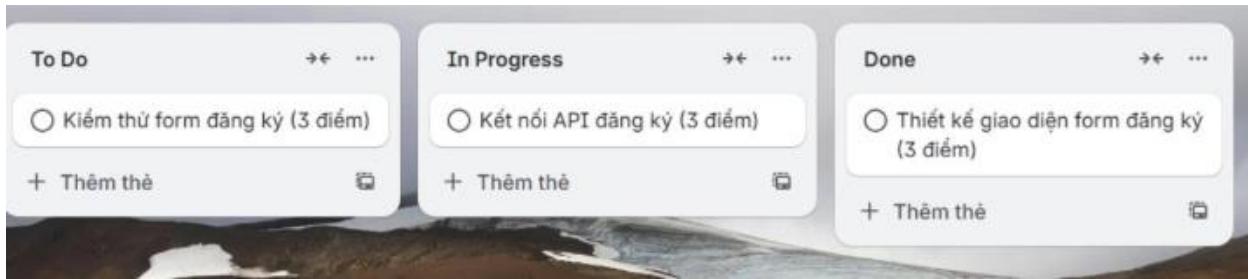
Bước 5: Vai trò Scrum Master

- Theo dõi tiến độ qua Kanban Board và Burndown Chart.
- Giải quyết trở ngại cho các thành viên.
- Đảm bảo nhóm tuân thủ Scrum và hoàn thành tasks trong Sprint.
- Hỗ trợ phối hợp giữa các thành viên, điều chỉnh kế hoạch nhỏ nếu cần.

Phản hồi ngắn

- **Điểm tốt hôm nay:** Nhóm hoàn thành đúng 3 task trong ngày, phối hợp hiệu quả.
- **Khó khăn:** Thiếu dữ liệu dummy và xác nhận endpoint API.
- **Scrum Master hỗ trợ:** Cung cấp dữ liệu test tạm thời, liên hệ backend để xác nhận API, giữ nhịp họp ngắn và hiệu quả.

Bài 8:



Daily Scrum – Buổi sáng (Mô phỏng theo 3 câu hỏi chuẩn)

Dev A – Frontend

- **Hôm qua đã làm gì?** Sprint bắt đầu hôm nay, chưa có công việc nào hoàn thành.
- **Hôm nay sẽ làm gì?** Thiết kế giao diện form đăng ký (UI).
- **Trở ngại gặp phải?** Cần Product Owner xác nhận các trường dữ liệu bắt buộc (email, mật khẩu, xác nhận mật khẩu).

Dev B – Backend

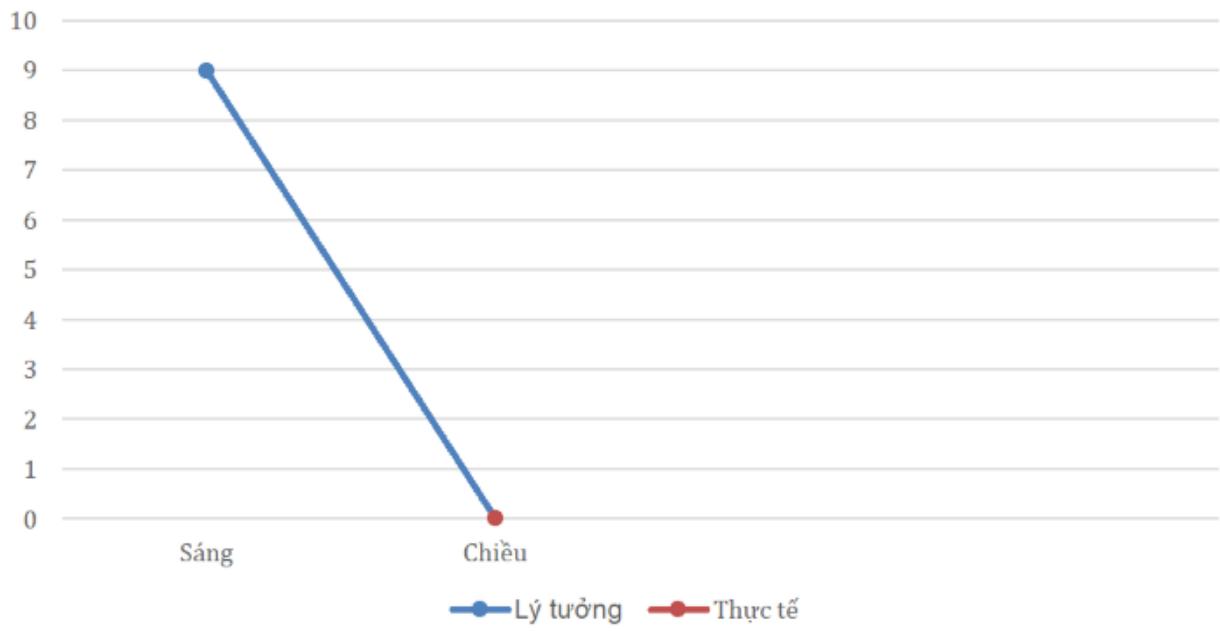
- **Hôm qua đã làm gì?** Không có, Sprint mới bắt đầu.
- **Hôm nay sẽ làm gì?** Kết nối API đăng ký và triển khai logic xác thực phía server.
- **Trở ngại gặp phải?** Chờ giao diện từ Dev A để đảm bảo request test đúng định dạng.

Tester C

- **Hôm qua đã làm gì?** Không có.
- **Hôm nay sẽ làm gì?** Viết test case cho form đăng ký.
- **Trở ngại gặp phải?** Chưa có bản build UI và API để thực hiện kiểm thử thực tế.

Burndown Chart – Cuối ngày

Burndown Chart



- **Tổng điểm Sprint hôm nay:** 9/9 điểm.
- **Công việc hoàn thành:**
 - Thiết kế UI: Hoàn thành.
 - Kết nối API: Hoàn thành.
 - Kiểm thử form: Hoàn thành.
- **Story Points còn lại:** 0/9.

Nhận xét: Sprint tiến triển đúng kế hoạch, các công việc liên quan UI → API → Testing được hoàn tất trong ngày.

Vai trò Scrum Master – Trong ngày

- Tổ chức Daily Scrum đúng thời gian (15 phút), đảm bảo mọi thành viên tập trung vào 3 câu hỏi chuẩn.
- Nhận diện nhanh các **impediments**: thiếu yêu cầu UI, API chưa có dữ liệu để test, tester chưa có bản build.
- Phối hợp với Product Owner để bổ sung yêu cầu và dữ liệu cần thiết ngay buổi sáng.
- Điều phối luồng công việc **UI → API → Testing**, đảm bảo tiến trình mượt mà, không gây tắc nghẽn.
- Giám sát và nhắc nhở team cập nhật **Kanban Board** liên tục, giữ minh bạch tiến độ công việc.

Bài 9 – Phân tích và giải pháp Scrum

1. Phân tích vấn đề

(A) Daily Scrum

- **Không bám Sprint Goal:** Team chỉ báo cáo hôm qua/hôm nay, nguy cơ làm sai hướng phát triển.
- **Không nêu impediments:** Velocity thấp nhưng vấn đề bị che giấu, ảnh hưởng đến Sprint Goal.
- **Daily kéo dài 22–25 phút:** Hẹp sai mục tiêu, tốn thời gian, giảm hiệu quả.

(B) Kanban Board tắc nghẽn

- Cột In Progress: 8 task (quá tải).
- Cột Done: chỉ 2 task → Team “bắt đầu nhiều – kết thúc ít”.
- **Không có WIP limit:** dẫn đến đa nhiệm, chậm tiến độ.

(C) Burndown Chart gần như đứng yên

- 5 ngày đầu: 52 → 51 → 51 → 48 → 47.
- Task không hoàn thành → Sprint Goal có nguy cơ thất bại.

2. Đề xuất can thiệp của Scrum Master

Vấn đề	Hành động Scrum Master	Hỗ trợ
Không bám Sprint Goal	Nhắc Sprint Goal mỗi Daily; Training team liên kết công việc với Goal	Dev Team + PO
Không nêu impediments	Giải thích thế nào là impediment; tạo môi trường nói thật	Dev + QA
Daily quá dài	Nhắc team chỉ trả lời 3 câu hỏi; chặn thảo luận kỹ thuật để sau	Trong cuộc họp
Kanban tắc nghẽn	Đặt WIP Limit (tối đa 3 task In Progress); ưu tiên kéo task sang Done	Dev + QA
Burndown không giảm	Kiểm tra task quá lớn → chia nhỏ; nhắc team cập	Dev Team

	nhật Kanban thường xuyên	
--	--------------------------	--

3. Mô phỏng Daily Scrum – Ngày thứ 6

Scrum Master: “Chúng ta bắt đầu Daily. Mỗi người trả lời: hôm qua – hôm nay – cản trở.”

Dev A: “Hôm qua em làm UI chọn bác sĩ. Hôm nay hoàn thiện UI chọn thời gian. Cản trở: API trả dữ liệu chậm.”

Dev B: “Hôm qua em làm API tạo lịch. Hôm nay làm API nhắc lịch. Không có trở ngại.”

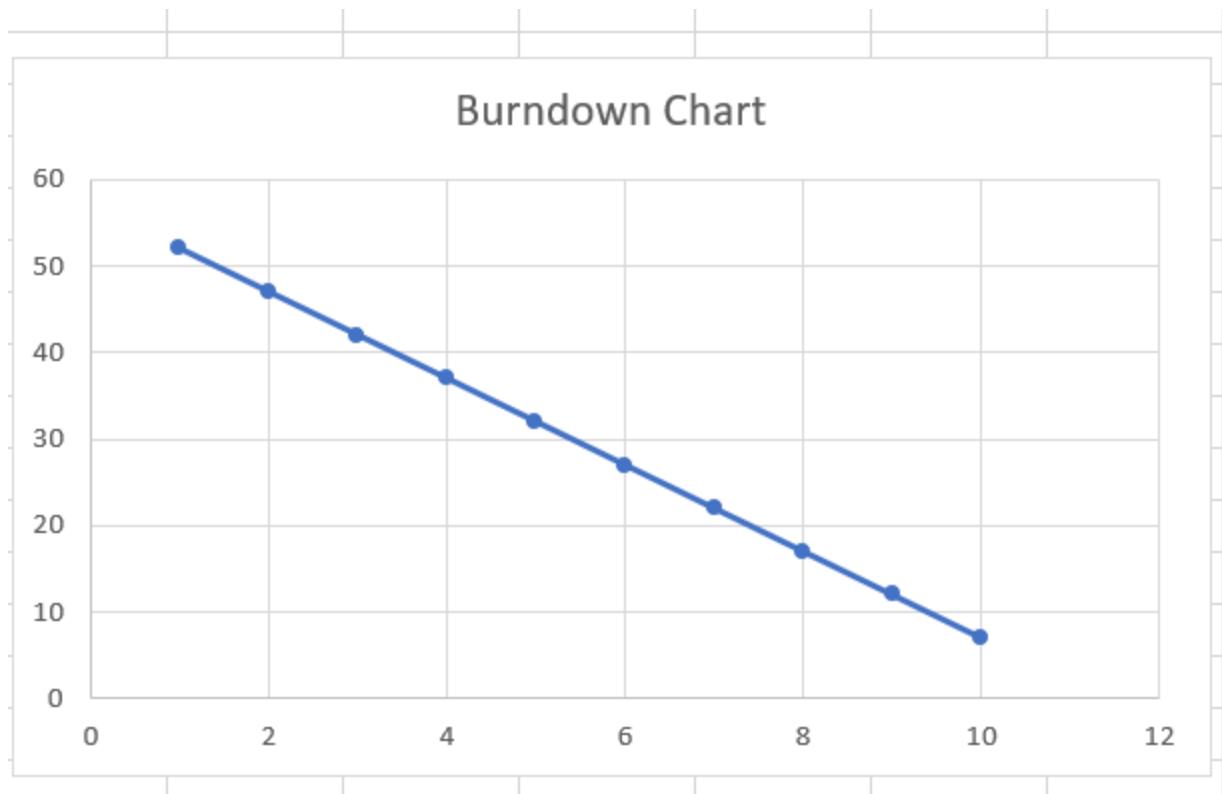
Dev C: “Hôm qua em làm lưu thông tin bệnh nhân. Hôm nay em test luồng đặt lịch. Cản trở: thiếu dữ liệu test.”

QA: “Hôm qua em viết test case. Hôm nay em test đặt lịch. Cản trở: chưa có bản build mới.”

Scrum Master: “Ghi nhận impediments. Chúng ta họp nhanh sau Daily để xử lý.”

4. Burndown Chart

- **Tổng Story Points:** 52, Sprint 10 ngày → giảm đều ~5 điểm/ngày.



Ý nghĩa:

- Giúp team biết cần burn bao nhiêu điểm mỗi ngày.
- So sánh với đường thực tế để phát hiện chậm tiến độ.
- Cảnh báo sớm, cho phép điều chỉnh để đảm bảo Sprint Goal đạt được.