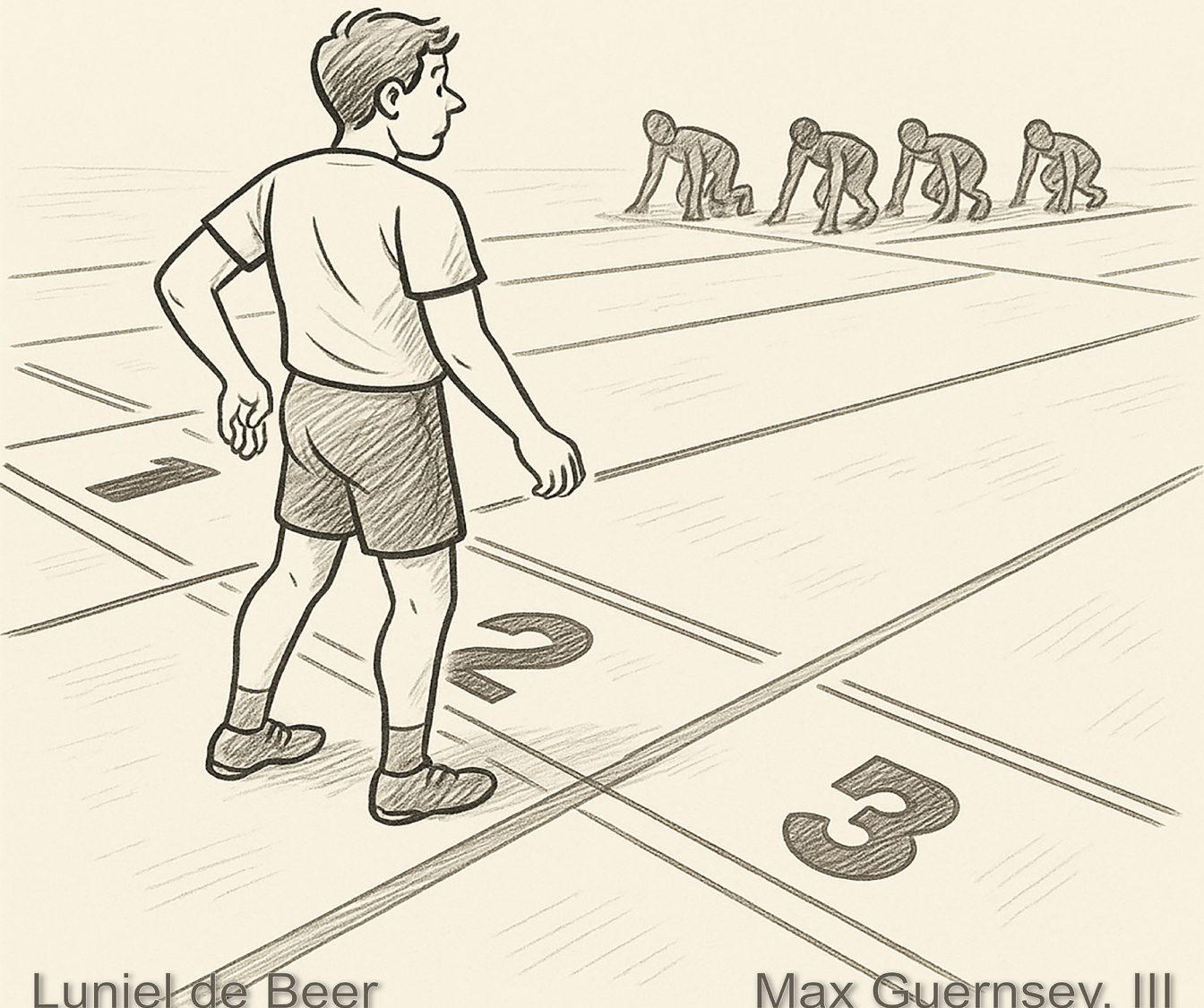


READY

WHY MOST SOFTWARE PROJECTS
FAIL AND HOW TO FIX IT



Luniel de Beer

Max Guernsey, III

Phiên Bản Tiếng Việt

Tweet về cuốn sách này!

Hãy giúp Luniel de Beer và Max Guernsey, III bằng cách chia sẻ về cuốn sách này trên [Twitter!](#)

Tweet được đề xuất cho cuốn sách này là:

Tôi vừa mua Ready — một cuốn sách dành cho các nhà lãnh đạo và đội ngũ phần mềm muốn loại bỏ việc làm lại, tồn đọng và sự không đồng bộ trong việc phát triển phần mềm. [#CodeReady](#)

Hashtag được đề xuất cho cuốn sách này là [#CodeReady](#).

Tìm hiểu những gì mọi người đang nói về cuốn sách bằng cách nhấp vào liên kết này để tìm kiếm hashtag này trên Twitter:

[#CodeReady](#)

Ready (Phiên Bản Tiếng Việt)

Tại Sao Đa Số Dự Án Phần Mềm Thất Bại và Cách Khắc Phục

Luniel de Beer và Max Guernsey, III

Cuốn sách này có sẵn tại <https://leanpub.com/ready-vi>

Phiên bản này được xuất bản vào 2025-10-22



Đây là cuốn sách trên [Leanpub](https://leanpub.com/ready-vi). Leanpub trao quyền cho các tác giả và nhà xuất bản thông qua quy trình Lean Publishing. [Lean Publishing](https://leanpub.com/ready-vi) là hành động xuất bản sách điện tử đang trong quá trình hoàn thiện bằng cách sử dụng các công cụ đơn giản và nhiều lần lặp lại để nhận phản hồi từ độc giả, điều chỉnh cho đến khi có được cuốn sách phù hợp và xây dựng sức hút khi đạt được điều đó.

© 2025 Luniel de Beer và Max Guernsey, III

Tuồng nhớ Johann van Aardt, người đã nhận ra niềm đam mê của tôi, giới thiệu tôi đến với lập trình thực thụ, và giúp tôi tìm được ngôi nhà mới. Và gửi đến cha mẹ tôi, những người với sự ủng hộ không lay chuyển—từ khách hàng đầu tiên cho đến ngôi nhà đầu tiên của tôi ở Hoa Kỳ—đã giúp tất cả điều này thành hiện thực.

—Luniel

Cho gia đình tôi, những người là lẽ sống của đời tôi.

—Max

Mục lục

Giới thiệu về Cuốn sách	i
Đối tượng Độc giả	ii
Cách Sử dụng Cuốn sách	iii
Về Tác giả	iv
Lời mở đầu	v
 Phần I: Điều Gì Đó Còn Thiếu	 1
Chương 1: Vấn đề Tiềm ẩn	2
Chương 2: Chi phí của việc thiếu nền tảng	14
Chương 3: Giới thiệu Requirements Maturation Flow (RMF)	26
Chương 4: Liệu Có Phải Là Agile?	28
 Phần II: Tạo Điều Kiện cho Sự Sẵn Sàng	 29
Chương 5: Phần Mở Rộng Đầu Tiên	30
Chương 6: Tại Sao Mọi Người Không Làm Điều Này?	31
Chương 7: Công việc Sẵn sàng Rõ ràng (RMF 1)	33
Chương 8: Các tác động của RMF 1	35
Chương 9: Đưa RMF 1 Vào Thực Tiễn	36

Phần III: Kiểm soát Hoàn thành Công việc 38

Chương 10: Nhu Cầu Tiếp Theo 39

Chương 11: Mọi Người Thường Làm Gì 41

Chương 12: Xác định Định nghĩa Hoàn thành 43

Chương 13: Định nghĩa Hoàn thành Tùy chỉnh (RMF 2) 45

Chương 14: Cuộc sống Với RMF 1 & 2 49

Chương 15: Cài đặt RMF 2 51

Phần IV: Triển khai có kiểm soát 53

Chương 16: Yêu cầu Cuối cùng 54

Chương 17: Bối cảnh về Định nghĩa về Sự sẵn sàng 56

Chương 18: Xác định Định nghĩa về Sự sẵn sàng 58

Chương 19: Định nghĩa Sẵn sàng Tùy chỉnh (RMF 3) 60

Phần V: Tổng hợp 64

Chương 20: Hầu hết Thời hạn Điều Không Quan trọng 65

Chương 21: Năng lực 1: Quy trình Hoàn thiện Yêu cầu 67

Chương 22: Cách Công việc và Thông tin Vận hành trong Scrum với RMF 69

Chương 23: Tác động của RMF 71

Chương 24: Chuyển đổi sang RMF 72

Chương 25: Phụ thuộc vào Bạn 74

Phần VI: Tài nguyên 75

Phụ lục A: Scrum Không Phải Là Vấn Đề 76

Phụ lục B: Khuôn khổ Synapse™ 78

Phụ lục C: Những Phản Đối và Trò Ngại Phổ Biến đối với RMF 1 80

Phụ lục D: Danh sách Tiêu chí Khởi đầu của DoD 81

Phụ lục E: Danh sách Tiêu chí Khởi đầu cho DoR 82

Mục lục 83

Giới thiệu về Cuốn sách

Ready là cuốn sách dành cho bất kỳ ai tham gia vào phát triển phần mềm và đang cảm thấy mệt mỏi với **việc không hoàn thành đủ yêu cầu, công việc phải làm lại thường xuyên, và yêu cầu không rõ ràng.**

Có thể bạn đã từng thử đầu tư vào việc nâng cao kỹ năng thực thi của nhóm, cải thiện việc thực hiện quy trình, hoặc nâng cấp mã nguồn nhưng vẫn cần cải thiện thêm.

Điều này là do ràng buộc chính đối với hầu hết các nhóm phát triển phần mềm không phải là kỹ năng nhóm, mà là mức độ hoàn thiện của yêu cầu. **Ngay cả những nhóm có kinh nghiệm với đầy đủ kỹ năng vẫn gặp khó khăn** khi họ làm việc với những yêu cầu chưa hoàn thiện.

Ready giới thiệu RMF (Quy trình Hoàn thiện Yêu cầu), một phương pháp thực tế và có cấu trúc sâu sắc để đồng bộ hóa Sản phẩm và Kỹ thuật mà không cần thay thế quy trình hiện có của bạn.

Cho dù bạn đang sử dụng Scrum, Kanban, hay một quy trình tùy chỉnh nào đó, RMF giúp bạn **ổn định phạm vi, loại bỏ việc kéo dài, và triển khai những gì thực sự quan trọng.**

Nếu nhóm của bạn cảm thấy bị mắc kẹt ở trạng thái “gần như hoàn thành”, cuốn sách này sẽ chỉ cho bạn cách **phá vỡ vòng luẩn quẩn** và giải phóng nhóm của bạn **một cách triệt để.**

Đối tượng Độc giả

Cuốn sách này thực sự dành cho bất kỳ ai tham gia vào phát triển phần mềm. Từ kỹ sư đến quản lý sản phẩm và từ nhân viên đến các nhà điều hành.

Cuốn sách này dành cho bạn nếu bạn tham gia vào phát triển phần mềm và nhận thấy rằng một nhóm mà bạn làm việc cùng hoặc trong đó có một hoặc nhiều vấn đề sau:

- Công việc thường xuyên bị kéo dài từ chu kỳ này sang chu kỳ khác
- Các nhóm thực hiện cảm thấy như đang cố gắng đuổi theo mục tiêu di động
- Công việc bị giữ mở quá lâu
- Công việc được đánh dấu là đã hoàn thành nhưng thực sự chưa xong
- Công việc hoàn thành không đáp ứng được kỳ vọng
- Công việc thường xuyên tạo ra nhiều lỗi
- Một lượng lớn công việc phải làm lại thường xuyên

Nếu bất kỳ vấn đề nào trong số đó nghe quen thuộc, *Ready* có thể giúp bạn.

Cách Sử dụng Cuốn sách

Cuốn sách này được thiết kế để mang tính thực tiễn. Nó không phải là một luận văn lý thuyết hay một bản thuyết trình chiến lược—mà là một hướng dẫn chi tiết về việc triển khai RMF (Quy trình Hoàn thiện Yêu cầu) dựa trên công việc thực tế với khách hàng và đã được kiểm chứng dưới áp lực triển khai thực tế.

Các chương được viết theo trình tự, nhưng bản thân RMF có tính module. Nó bao gồm ba thực hành nền tảng:

- RMF 1: Hợp tác để đạt được sự hiểu biết chung
- RMF 2: Kiểm soát việc hoàn thành công việc bằng Định nghĩa Hoàn thành Tùy chỉnh
- RMF 3: Kiểm soát việc thực hiện bằng Định nghĩa Sẵn sàng Tùy chỉnh

Mỗi phần, hay “Thói quen” như chúng tôi gọi, có thể đứng độc lập, nhưng chúng xây dựng dựa trên nhau. Cuốn sách được thiết kế để giúp bạn tiếp cận chúng từng cái một, theo thứ tự. Cấu trúc đó phản ánh cách chúng tôi khuyến nghị các nhóm áp dụng RMF trong thực tế—với mỗi Thói quen chỉ được đưa vào sau khi thói quen trước đó đã hoạt động tốt.

Điều này tránh làm quá tải các nhóm và cho mỗi thay đổi cơ hội tốt nhất để bám rễ. Bạn sẽ học thêm về cách thực hiện điều đó bắt đầu từ [Chương 9](#).

Nếu bạn đang tìm kiếm sự trợ giúp—cho dù đó là lời khuyên, huấn luyện, hay ai đó để nói chuyện với đội ngũ lãnh đạo của bạn—hãy thoải mái liên hệ trực tiếp với chúng tôi.

Và nếu bạn đang tìm kiếm hỗ trợ chính thức để triển khai RMF, Producore cung cấp một loạt Chương trình đầy đủ được thiết kế để hướng dẫn việc áp dụng từng bước. Bạn có thể tìm hiểu thêm tại <https://ready-book.link/rmf>.

Về Tác giả

Luniel de Beer là người sáng tạo ra Quy trình Hoàn thiện Yêu cầu (RMF), một hệ thống thực tế để khắc phục khoảng cách giữa ý định sản phẩm và việc thực thi kỹ thuật. Anh có hơn 15 năm kinh nghiệm dẫn dắt các chuyển đổi Agile, kết nối sản phẩm và kỹ thuật, và giúp các nhóm triển khai với sự rõ ràng và tự tin.

Luniel cũng là người khởi xướng hệ thống Quản lý Năng lực của Producure, một phương pháp có thể theo dõi và mở rộng để mô hình hóa các năng lực sản phẩm. Anh đã hình dung ra Phát triển Dựa trên PKB (PKBDD), một hệ thống kiểm soát phiên bản để quản lý các yêu cầu sản phẩm liên tục. Những công cụ này tạo thành một phần của khung triển khai lớn hơn được phát triển tại Producure.

Max Guernsey, III là một kiến trúc sư phần mềm, nhà giáo dục, và đồng sáng lập của Producure, một công ty tư vấn chuyên khắc phục các thất bại trong triển khai thông qua sự nghiêm ngặt về cấu trúc và kỹ thuật. Với hơn hai thập kỷ kinh nghiệm trong thiết kế hướng đối tượng, tái cấu trúc, phát triển hướng kiểm thử, và mẫu thiết kế, anh đã triển khai các hệ thống quan trọng và huấn luyện các nhóm kỹ thuật ở quy mô lớn. Công việc của anh kết hợp các thực hành kỹ thuật sâu sắc với sự chuyển đổi về hành vi và quy trình để giúp các tổ chức đạt được sự xuất sắc bền vững trong triển khai.

Max đã đóng góp đáng kể cho PKBDD và dẫn dắt việc phát triển phương pháp Phát triển hướng hành vi (BDD) của Producure thông qua chuyên môn sâu rộng của anh về đặc tả hành vi.

Cùng nhau, công việc của họ tích hợp tính rõ ràng, khả năng truy vết, và kiểm soát cửa vào một hệ thống mạch lạc cho việc triển khai phần mềm, có khả năng mở rộng từ thực hành nhóm đến năng lực tổ chức.

Lời mở đầu

Lưu ý cho các Nhà Lãnh đạo Kỹ thuật

Nếu bạn là một lãnh đạo cấp cao trong tổ chức kỹ thuật, có lẽ bạn không thiếu nỗ lực, kỷ luật, hay người tài. Tuy nhiên, không hiểu sao, các dự án vẫn bị đình trệ. Mục tiêu bị trượt. Kỳ vọng không được đáp ứng. Không phải vì đội ngũ của bạn lười biếng—mà bởi vì có điều gì đó căn bản đã bị hỏng trong cách công việc được định nghĩa, định hình và triển khai.

Cuốn sách này không phải là một hướng dẫn về lãnh đạo. Nó là một công cụ để chẩn đoán cấu trúc. Nó cho thấy điều gì thực sự đang diễn ra bên trong các đội của bạn—tại sao “gần như hoàn thành” cứ biến thành “chưa hoàn thành,” và tại sao tiến độ cục bộ hiếm khi chuyển thành kết quả chiến lược.

Bạn có thể không thấy *bản thân* mình trong những trang này. Nhưng nếu các đội của bạn không thể triển khai những gì bạn cần, bạn sẽ thấy *họ*. Và khi điều đó xảy ra, cuối cùng bạn sẽ có ngôn ngữ—và hệ thống—để sửa chữa nó.

Từ Luniel

Trước hết, cuốn sách này sẽ không thể thành hiện thực nếu không có Max, người có khả năng nhìn xuyên qua sương mù và nhiễu loạn để chốt lọc một ý tưởng thành bản chất của nó, điều mà tôi không thể làm được.

Làm sao chúng tôi đến được đây?

Nếu nhìn lại, tôi nghĩ đó là vì tôi luôn muốn hiểu mọi thứ thực sự hoạt động như thế nào. Dù đó là tôn giáo, dinh dưỡng, hay phát triển phần mềm, tôi luôn gặp phải cùng một vấn đề: những câu trả lời hời hợt không đứng vững trước áp lực. Vì vậy tôi tiếp tục đào sâu—không chỉ hỏi chúng ta làm gì, mà còn tại sao, và điều gì còn thiếu khi nó không hoạt động.

Một trong những rạn nứt đầu tiên trong hệ thống xuất hiện khi tôi đảm nhận ba vai trò: Scrum Master, Product Owner, và Development Manager (!) cho một đội phát triển dịch vụ dữ liệu tại một công ty công nghệ nổi tiếng. Chúng tôi đang làm những gì Scrum yêu cầu—Sprint ngắn, các story trong backlog, lập kế hoạch trong nửa ngày—nhưng mỗi khi bắt

đầu một Sprint mới, chúng tôi lại gặp ma sát. Đội không hiểu rõ vấn đề, chúng tôi phải xem xét và sửa đổi yêu cầu giữa Sprint, các phụ thuộc có thể phòng tránh được lại xuất hiện và làm chậm trễ chúng tôi, và các bước quan trọng bị bỏ sót.

Vì vậy tôi bắt đầu làm điều gì đó khác. Tôi tập hợp đội và các bên liên quan vào một phòng cho mỗi story, cùng nhau xem xét chi tiết vấn đề, động não về giải pháp, và chỉ sau đó mới viết story. Sprint Planning giảm xuống còn một giờ, và sự thành công trong việc triển khai của chúng tôi tăng vọt.

Dần dần, tôi bắt đầu nhận ra rằng thành công không đến từ việc làm việc chăm chỉ hơn trong Sprint. Nó đến từ cấu trúc bạn thiết lập *trước khi* nó bắt đầu.

Sau đó, sau khi nghe Jeff Sutherland nói về “Definitions of Ready”, tôi biết có điều gì đó có giá trị ở đó—nhưng nó chưa đủ. Kinh nghiệm của tôi với yêu cầu, UX, UI, nghiên cứu, và sau đó với BDD cho tôi thấy rằng các hạng mục công việc khác nhau đòi hỏi các loại sẵn sàng khác nhau. Một số cần đặc tả hành vi. Một số cần quyền truy cập hệ thống. Một số cần truy xuất năng lực đầy đủ.

Và tất cả đều cần sự hiểu biết chung thực sự được xác nhận—không phải giả định.

Khi làm việc với nhiều đội hơn, tôi thấy cùng một khuôn mẫu ở khắp mọi nơi: các bước bị bỏ sót, các phụ thuộc không được đáp ứng, các đội làm hết sức nhưng liên tục phải vật lộn để sửa những vấn đề lẽ ra có thể ngăn ngừa được. Ngay cả những đội giỏi cũng gặp khó khăn—không phải vì họ yếu, mà vì họ thiếu một cấu trúc làm cho sự sẵn sàng trở nên rõ ràng.

Kết quả của tất cả việc học hỏi, cải tiến và thất vọng đó là một hệ thống có cấu trúc để quản lý sự sẵn sàng.

Đó là nội dung của cuốn sách này.

Tôi hy vọng nó sẽ giúp bạn hiểu rõ về nơi các vấn đề thực sự nằm ở đâu, và cách khắc phục chúng. Tôi hy vọng nó sẽ cung cấp cho bạn ngôn ngữ để bảo vệ các thực hành có vẻ “thừa” nhưng thực ra là thiết yếu. Và quan trọng nhất, tôi hy vọng nó sẽ giúp các đội triển khai với ít căng thẳng hơn, ít bất ngờ hơn, và kết quả tốt hơn nhiều.

Nếu chúng ta làm đúng điều này, chúng ta sẽ tiết kiệm được hàng tỷ đô la cho ngành công nghiệp.

Nhưng quan trọng hơn, chúng ta sẽ giúp mọi người lấy lại sự tinh tảo của họ.

Từ Max

Tôi đã làm việc với vấn đề này từ nhiều góc độ khác nhau trong nhiều thập kỷ, nhưng tiến độ của tôi bị đình trệ cho đến khi tôi gặp Luniel.

Điều này là vì, trước khi tôi biết anh ấy, về cơ bản tôi đã tiếp cận vấn đề như một vấn đề kỹ thuật. Tôi tập trung vào việc giúp các đội áp dụng những thứ như Test-Driven Development (TDD), refactoring, thiết kế phần mềm nâng cao và, sau đó là Acceptance-Test-Driven Development (ATDD) hoặc Behavior-Driven Development (BDD).

Trong hầu hết các trường hợp đó, vấn đề được đề cập trong cuốn sách này được coi là một chi tiết triển khai của việc thiết lập các thực hành kỹ thuật đó.

Điều này không có nghĩa là tôi không còn coi trọng các thực hành kỹ thuật. Tôi vẫn nghĩ chúng rất quan trọng, nhưng chúng không trực tiếp giải quyết vấn đề về sự sẵn sàng trong phát triển phần mềm. Thay vào đó, chúng *làm nổi bật* vấn đề đó và sau đó mọi người vá vúi quy trình của họ để giải quyết nó “vừa đủ” để hỗ trợ các thực hành kỹ thuật mà họ đang cố gắng thực hiện.

Tôi cũng muốn đề cập đến câu hỏi cuốn sách này có thể giúp ai. Câu trả lời ngắn gọn là “có lẽ gần như tất cả mọi người trong phát triển phần mềm”, nhưng câu trả lời thực sự chứa đựng những sắc thái giúp ánh xạ nó vào các môi trường khác nhau mà không thay đổi ý nghĩa cơ bản.

Có những đội cần đến giải pháp được trình bày trong cuốn sách này. Bạn sẽ được gặp một phiên bản đã được làm gọn của một đội như vậy trong Chương 1.

Cũng có những đội không thực sự cần một hệ thống như chúng tôi đề xuất, nhưng vẫn có thể hưởng lợi từ nó.

Đội xuất sắc nhất mà tôi từng làm việc cùng—dễ dàng vượt trội hơn **đội xếp sau** một đơn vị độ lệch chuẩn, nếu không muốn nói là hai—được đặt tại vùng hẻo lánh của Central Oregon. Họ có hiệu suất cao đến mức có thể vượt qua được sự thiếu vắng của một hệ thống như vậy bằng năng lực vượt trội của mình. Tuy nhiên, người quản lý của tôi lúc đó, Tom Barreras, đã từng nói với tôi điều gì đó đại loại như “Tôi nhận thấy rằng các câu chuyện của chúng ta diễn ra tốt hơn khi chúng ta dành thời gian thảo luận về các bài kiểm thử ngay từ đầu.”

Một lần nữa, đây là điều mà lúc đó tôi chỉ nhìn nhận qua lăng kính của phát triển kiểm thử và việc thực thi kỹ thuật, nhưng giờ đây tôi biết đó là một dấu hiệu khác cho thấy *sự sẵn sàng* là một yếu tố ảnh hưởng đến đội... đội đặc biệt đó giỏi và phản ứng nhanh đến mức họ có thể thành công bằng cách đối phó với các trở ngại khi chúng xảy ra thay vì phòng ngừa chúng ngay từ đầu.

Ngay cả khi bạn là kiểu người không nhất thiết phải *lo lắng* về sự sẵn sàng vì bạn có thể vượt qua được nó, hoặc bạn làm việc trong một đội có cùng tính cách như vậy, bạn vẫn có thể hưởng lợi từ nội dung của cuốn sách này.

Phần I: Điều Gì Đó Còn Thiếu

*Khi việc làm tốt hơn những việc cũ không giúp ích được gì, hãy tìm kiếm **những điều** chưa được thực hiện.*

Chương 1: Vấn đề Tiềm ẩn

Đây là một câu chuyện có thật¹ về một ngân hàng. Chúng ta sẽ gọi nó đơn giản là “Ngân hàng”. Đây là một loại Tổ chức Tín dụng Liên bang đóng vai trò như một phần của cơ sở hạ tầng tài chính quốc gia Hoa Kỳ.

Chúng tôi (Luniel và Max) được mời đến Ngân hàng vì họ đang gặp khó khăn trong việc triển khai một dự án phần mềm. Đây là một trong những môi trường làm việc rối loạn nhất mà chúng tôi từng thấy, và đó là lý do tại sao chúng tôi chọn đây làm nghiên cứu điển hình mở đầu: nếu sự thay đổi có ý nghĩa là có thể xảy ra tại Ngân hàng này, thì nó có thể xảy ra ở bất cứ đâu.

Một Ghi chú Nhanh về Dự án

Khi chúng tôi sử dụng thuật ngữ “dự án” trong cuốn sách này, chúng tôi đề cập đến nó trong bối cảnh quản lý dự án. Mặc dù có nhiều ý kiến khác nhau về ý nghĩa của từ này, chúng tôi sử dụng định nghĩa từ [Project Management Institute](#):

“Một dự án là một nỗ lực **tạm thời** được thực hiện để tạo ra một sản phẩm, dịch vụ hoặc kết quả độc đáo.”

Điều này có nghĩa là một dự án có điểm bắt đầu và kết thúc được xác định. Khi một dự án được đóng lại, kiến thức và các tài liệu dự án được lưu trữ, các thành viên nhóm được giải phóng, và các hợp đồng được hoàn tất.

Trong cuốn sách này, một dự án về cơ bản là về việc thực thi. Hầu hết các dự án, theo định nghĩa của PMI, bắt đầu với tính khả thi hoặc thiết kế. Tầm nhìn và chiến lược đã được xác định trước khi một dự án được khởi động.

Một dự án được sinh ra từ tầm nhìn và chiến lược đó, và nó thành công hay thất bại dựa trên việc liệu các mục tiêu đã định có được thực hiện hay không—*không phải* dựa trên việc liệu những mục tiêu đó có đúng đắn hay không.

¹Chúng tôi đã thay đổi các chi tiết nhận dạng để bảo vệ quyền riêng tư của những người và tổ chức liên quan.

Bạn có thể sử dụng từ “dự án” theo cách khác, và điều đó không sao cả. Chỉ cần biết rằng khi chúng tôi sử dụng nó, chúng tôi đang đề cập đến định nghĩa và bối cảnh ở trên.

Không có điều nào trong số này ngụ ý rằng chúng tôi ủng hộ việc sử dụng quản lý dự án cho phát triển phần mềm. Hoàn toàn ngược lại. Nhưng chúng tôi nhận ra rằng nó *vẫn* được sử dụng. Chúng tôi sẽ giải quyết vấn đề đó sau, trong [Chương 6](#).

1.1: Việc Viết lại Cổ điển

Ngân hàng đang viết lại cổng thanh toán khoản vay của họ vì một số lý do.

Hệ thống cũ, một giải pháp C#/.NET hoàn chỉnh, có nhiều lỗi. Ngoài việc làm giảm sự hài lòng của khách hàng, nó còn tạo ra một dòng không ngừng các sự cố hỗ trợ rất tốn kém, trong đó ai đó phải thao tác thủ công với cơ sở dữ liệu để sửa lỗi do hệ thống gây ra.

Hệ thống cũ cũng đã lỗi thời từ góc độ khả năng bảo trì. Gần như không thể để các kỹ sư thực hiện những thay đổi có ý nghĩa và, ngay cả khi họ có thể, đó cũng là một việc làm cực kỳ rủi ro.

Việc viết lại được cho là sẽ thay đổi điều đó.

Hệ thống mới vẫn sẽ có backend C#/.NET, nhưng sẽ được bao phủ hoàn toàn bởi các bài kiểm thử. Frontend sẽ được triển khai bằng OutSystems, một giải pháp low-code hoặc no-code phổ biến cho phép một tổ chức định nghĩa ứng dụng tại một nơi và tự động tạo ra ứng dụng web, ứng dụng Android và ứng dụng iOS bất cứ khi nào họ quyết định xuất bản các thay đổi của mình.

Họ hy vọng rằng hệ thống mới sẽ không có lỗi, vừa cải thiện sự hài lòng của khách hàng vừa giảm đáng kể chi phí hỗ trợ.

Họ cũng hy vọng rằng việc viết lại sẽ giải phóng các nhà phát triển—with sự kết hợp giữa cách tiếp cận có kỷ luật hơn đối với backend và cách tiếp cận low-code đối với frontend sẽ giảm đáng kể chi phí và rủi ro của các tính năng mới.

Một tác dụng phụ tốt của việc chuyển sang OutSystems là họ sẽ có một ứng dụng di động sạch sẽ, hiện đại trên cả hai nền tảng chính.

Đó là giấc mơ khi họ bắt đầu ba năm trước khi câu chuyện này bắt đầu. Thực tế là, cho đến nay, các đội không triển khai được **bất cứ thứ gì**.

1.2: Các Góc nhìn về Vấn đề

Khi chúng tôi nói chuyện với đội ngũ lãnh đạo điều hành, chúng tôi nghe thấy sự thất vọng rất tự nhiên về việc họ đã đầu tư rất lớn mà hoàn toàn không có bất kỳ bước tiến chiến lược nào.

Theo quan điểm của họ, họ đã thử mọi cách. Họ đã thay đổi nhân sự, tăng nhân sự, thay đổi ngân sách, tăng áp lực, và mời về một loạt các chuyên gia tư vấn (trong đó ngụ ý rằng chúng tôi là những người cuối cùng). Không có gì dường như làm cho tình hình tốt hơn—ít nhất là không theo cách mà họ có thể đo lường được, bởi vì tất cả những gì họ thấy là “kim đồng hồ” ở mức không trong một quý và vẫn ở mức không trong quý tiếp theo.

Họ không muốn thêm bất kỳ “tiến độ vô hình” nào nữa. Họ muốn *kết quả*.

Khi chúng tôi nói chuyện với ban quản lý trong Tổ chức Sản phẩm, chúng tôi nhận được một câu chuyện hơi khác (nhưng vẫn tương tự) vì họ làm việc trực tiếp hơn với bộ phận Kỹ thuật.

Không phải là các nhóm không làm gì cả, ít nhất là theo quan điểm của họ. Mà là các nhóm không bao giờ làm những gì được yêu cầu. Điều này gần như có thể đảm bảo chắc chắn: dù yêu cầu đơn giản đến đâu và được trình bày rõ ràng thế nào, khi đến lúc đánh giá thành quả của các nhóm, bạn sẽ nhận được một thứ hoàn toàn khác.

Tình hình đã đến mức mà câu đùa quen thuộc là “Chúng ta cần tìm cách yêu cầu điều chúng ta không muốn, để ít nhất có cơ hội nhận được điều chúng ta thực sự muốn.”

Các giám đốc kỹ thuật lại nhìn nhận vấn đề hoàn toàn khác.

Đối với họ, vấn đề nằm ở chỗ bộ phận Sản phẩm không đưa ra các yêu cầu khả thi và không đưa ra *đủ* yêu cầu. Nếu bộ phận Sản phẩm có thể “đi đúng hướng”, các nhóm sẽ có thể cung cấp những gì họ muốn đúng thời hạn và trong ngân sách.

Họ đã đầu tư nghiêm túc vào việc hiện đại hóa cách viết và triển khai mã nguồn và, theo quan điểm của họ, bộ phận Sản phẩm không đưa ra yêu cầu rõ ràng.

Khi chúng tôi nói chuyện với các chuyên gia tư vấn khác (những người giới thiệu chúng tôi vào tổ chức), họ đã đúng khi chỉ ra sự rối loạn chức năng mà họ nhìn thấy: Mọi người dường như rất tập trung vào việc đổ lỗi cho người khác. Lý do họ mời chúng tôi vào từ đầu là vì họ lo ngại về chiến lược nhân sự và muốn đánh giá những người thực hiện, nhưng họ cho rằng việc chỉ trỏ đổ lỗi và tư duy quản lý kiểu giám sát ở cấp điều hành là nguồn gốc chính của vấn đề.

1.3: Cuộc Điều Tra Của Chúng Tôi

Nhiệm vụ ban đầu của chúng tôi là đánh giá các nhóm và cố gắng giúp họ nâng cao kỹ năng nếu cần thiết, vì vậy chúng tôi bắt đầu tìm hiểu về những người làm việc ở tuyến đầu.

Chắc chắn là có nhiều điều cần cải thiện.

Những người thực hiện ở phía Sản phẩm không thực sự có các kỹ năng cần thiết. Trong thực tế, họ phần lớn là các quản lý dự án được đẩy vào vai trò Chủ Sản Phẩm (PO) hoặc Quản Lý Sản Phẩm.

Kết quả là, một nửa trong số họ viết các yêu cầu “mơ hồ” và sau đó chấp nhận (theo nghĩa đen) bất cứ điều gì các nhóm làm trong sprint đó mà không có bất kỳ phân tích phê bình nào. Nửa còn lại viết cùng loại yêu cầu và sau đó tuyên bố rằng các nhóm “đáng lẽ phải biết” những thứ mà họ chưa bao giờ đề cập đến và giữ các hạng mục công việc mở vô thời hạn.

Ngân hàng đã sử dụng để quản lý và theo dõi danh sách công việc tồn đọng của họ. Mặc dù những gì chúng tôi đề cập trong cuốn sách này phần lớn độc lập với Scrum, chúng tôi sử dụng thuật ngữ Scrum xuyên suốt vì đa số - hoặc ít nhất là phần lớn - các nhóm sử dụng Scrum.



Định nghĩa: Scrum

Scrum là một khung làm việc nhẹ giúp mọi người, các nhóm và tổ chức tạo ra giá trị thông qua các giải pháp thích ứng cho các vấn đề phức tạp. Tóm lại:

1. Một Chủ Sản Phẩm sắp xếp công việc cho một vấn đề phức tạp vào Danh Sách Công Việc Sản Phẩm
2. Nhóm Scrum chuyển một phần công việc được chọn thành một phần giá trị trong một Sprint
3. Nhóm Scrum và các bên liên quan kiểm tra kết quả và điều chỉnh cho Sprint tiếp theo
4. Lặp lại

Nếu bạn không quen thuộc với Scrum và thuật ngữ của nó, chúng tôi khuyên bạn nên xem lại phiên bản 2020 của [Scrum Guide](#). Đó là một tài liệu ngắn gọn và sáng sủa.

Tương tự, chúng tôi nhận thấy các nhóm kỹ thuật có kỹ năng lập trình thấp hơn nhiều so

với mức thông thường (-2σ , trong trường hợp tốt nhất) và còn rất kháng cự với sự thay đổi. Hệ quả tự nhiên là chất lượng mã nguồn rất tệ.

Tuy nhiên, theo các nhóm, đây không phải là lý do tại sao họ không hoàn thành công việc. Đối với họ, chính các yêu cầu mơ hồ và những thay đổi giữa Sprint của bộ phận Sản phẩm đang phá hỏng dự án.

...và không ai thậm chí đề cập đến vấn đề lớn hơn, một vấn đề phi lý đến mức tưởng chừng được bịa ra cho đến khi bạn thực sự trải qua nó.

Các nhóm kỹ thuật có thói quen không hiểu một yêu cầu, xây dựng một thứ gì đó ngẫu nhiên, và sau đó đòi được ghi nhận là đã “hoàn thành một hạng mục công việc”.

Chúng tôi không nói về một sự hiểu nhầm nhỏ. Chúng tôi nói về một sự ngắt kết nối hoàn toàn: Chúng tôi bảo họ vô hiệu hóa việc áp dụng quỹ vào tiền gốc trong một số trường hợp nhất định, và họ lại vô hiệu hóa khả năng thêm địa chỉ email xác nhận phụ.

Sau đó họ nói với chúng tôi rằng đó là những gì chúng tôi đã yêu cầu.

1.4: Đào Sâu Hơn

Cả hai khoảng cách về kỹ năng đều có thể được giải quyết, nhưng chúng tôi hoài nghi liệu đó có phải là những rào cản thực sự hay không.

Có điều gì đó khác đang sai, vì vậy chúng tôi đào sâu hơn. Chúng tôi bắt đầu với câu hỏi này: Tại sao việc viết yêu cầu lại mất nhiều thời gian và tạo ra kết quả tệ như vậy?

Một lý do là kiến thức cần thiết để viết một yêu cầu có ý nghĩa rất khan hiếm. Một phần nhỏ trong số đó nằm ở các nhóm Kỹ thuật và Sản phẩm.

Một số nằm trong mã nguồn của hệ thống cũ. Một số đã biến mất hoàn toàn. Tuy nhiên, phần lớn được lưu trữ dưới dạng kiến thức nội bộ trong đầu các chuyên gia về lĩnh vực rải rác khắp các bộ phận của ngân hàng. Điều này có nghĩa là việc xây dựng một yêu cầu thực sự thúc đẩy mục tiêu chiến lược là một hoạt động cực kỳ tốn công sức và thời gian.

Đối lập với điều này là sự thèm khát không thể thỏa mãn về chức năng từ đội ngũ lãnh đạo đang thiếu tính năng. Chỉ thị là “giữ cho các kỹ sư làm việc—nhồi nhét họ đầy yêu cầu”. Trọng tâm là số lượng yêu cầu để giữ cho các nhóm bận rộn—một quan điểm đi ngược lại với sự cẩn trọng và thời gian cần thiết để định nghĩa một yêu cầu có thể “tạo ra sự khác biệt”.

1.5: Cải thiện Mọi Thứ Không Làm Cho Nó Tốt Hơn

Đây đều là những vấn đề có thể giải quyết được, nhưng việc giải quyết chúng không giúp ích gì.

Những cải tiến trước đây về kỹ thuật phát triển phần mềm đã không giúp các nhóm hoàn thành công việc, nhưng Max đã nỗ lực để giúp các nhóm cải thiện hơn nữa.

Anh ấy đã giới thiệu những khái niệm mang tính cách mạng từ các học thuyết lập trình giữa thế kỷ 20 như “đừng sao chép và dán đoạn mã đó 27² lần, hãy đặt nó vào một hàm và gọi hàm đó thay thế”. Chỉ riêng đề xuất này đã cải thiện đáng kể chất lượng mã môi và cho phép họ bắt đầu nâng cao chất lượng.

Lời khuyên lập trình cơ bản đó và những lời khuyên khác đã giúp họ viết mã tốt hơn mà họ có thể dễ dàng bảo trì trong tương lai.

...nhưng nó không giúp dự án tiến triển.

Về phía Sản phẩm, Luniel đã có thể giới thiệu BDD và đảm bảo rằng các Chủ sở hữu Sản phẩm kiểm tra kỹ lưỡng các yêu cầu trước khi chuyển giao cho các nhóm.

Anh ấy đã khiến các nhóm hợp tác về các yêu cầu và sử dụng chúng để đánh giá xem một Hạng mục Tồn đọng Sản phẩm (PBI) có thực sự hoàn thành hay không.



Định nghĩa: Hạng mục Tồn đọng Sản phẩm (PBI)

Hạng mục Tồn đọng Sản phẩm (PBI) là một đơn vị công việc riêng biệt trong tồn đọng sản phẩm, đại diện cho một thay đổi, bổ sung hoặc cải tiến tiềm năng cho sản phẩm. PBI có thể có nhiều hình thức—tính năng, sửa lỗi, cải tiến kỹ thuật, nhiệm vụ nghiên cứu, v.v.—và được xác định bởi đóng góp của chúng vào giá trị sản phẩm.

Nhiều nhóm gọi PBI là “câu chuyện” hoặc “câu chuyện người dùng,” nhưng thuật ngữ Scrum chính xác là “Hạng mục Tồn đọng Sản phẩm” hoặc “PBI.” Khi một PBI được cam kết vào Sprint, nó cũng là một phần của Tồn đọng Sprint. Để đơn giản và trung lập, chúng tôi sử dụng “Hạng mục Tồn đọng Sản phẩm” hoặc “PBI” để chỉ bất kỳ hạng mục công việc nào mà Nhóm Scrum đang quản lý—cho dù bạn gọi nó là Hạng mục Tồn đọng Sản phẩm, Hạng mục Tồn đọng Sprint (SBI), câu chuyện người dùng, câu chuyện, hạng mục công việc, hay hạng mục tồn đọng.

²Điều này không chỉ không phải là phóng đại, mà thậm chí còn không phải là trường hợp tệ nhất. Trong một trường hợp, có gần một trăm bản sao chính xác của cùng một thuật toán.

Nó tạo thêm *sự rõ ràng*, nhưng không tạo ra *dòng chảy*.

Với sự giúp đỡ của các chuyên gia tư vấn đối tác đã mời chúng tôi vào, chúng tôi đã có thể (tạm thời) giảm bớt áp lực cực kỳ nặng nề đặt lên các nhóm và người viết yêu cầu.

Điều đó có thể đã giúp tạo ra một chút niềm tin, nhưng không mang lại kết quả cụ thể nào.

Chúng tôi thậm chí đã bắt đầu xây dựng một cơ sở kiến thức giúp mọi người theo dõi kiến thức kinh doanh họ cần để viết yêu cầu và xác định những nơi có khoảng trống trong kiến thức đó.

Nó đã đẩy nhanh việc viết yêu cầu, nhưng không giúp đưa sản phẩm ra khỏi cửa.

Sau nhiều tháng tương tác, chúng tôi đã giúp ban lãnh đạo *nhìn thấy* họ đang ở đâu, nhưng họ vẫn chưa gần với mục tiêu của mình. Và họ cũng không tiến gần hơn chút nào.

Họ đang chuẩn bị quay trở lại chiến lược cũ là “nhồi nhét” các nhóm để đảm bảo họ luôn bận rộn.

1.6: Một Quá trình Loại trừ

Sẽ rất dễ dàng để chỉ giờ tay lên và nói “điều này là vô vọng”. Có rất nhiều lý do có thể viện đến:

- Đội ngũ kỹ sư có kỹ năng thấp (đúng vậy)
- Những người viết yêu cầu có bộ kỹ năng không phù hợp (đúng vậy)
- Ban lãnh đạo đang đè nặng các nhóm với những kỳ vọng phi thực tế (đúng vậy)
- Tổ chức đang thiếu kiến thức kinh doanh quan trọng cần thiết để hoạt động (đúng vậy)
- Các giám đốc điều hành dường như không tin tưởng lẫn nhau (đúng vậy)

Tất cả những điều đó đều đúng. Tuy nhiên, những cải tiến đã được thực hiện trong tất cả các lĩnh vực đó và không có cái nào dường như làm cho vấn đề cơ bản tốt hơn. Không có cái nào giúp các nhóm Kỹ thuật, Sản phẩm, ban quản lý, hoặc ban điều hành tiến gần hơn đến mục tiêu của họ.

...và đó chính là gợi ý cho giải pháp ngay tại đó: Tất cả các vấn đề được liệt kê trước đó là những vấn đề mà mọi người đã có thể nhìn thấy.



Nếu những biến số bạn có thể thấy không tạo ra sự khác biệt, thì chắc chắn phải có một biến số mà bạn **không nhìn thấy** đang tạo ra sự khác biệt đó.

Vấn đề thực sự nằm ở chỗ không ai biết đến sự tồn tại của nó.

1.7: Truy tìm thủ phạm thực sự

Trong quá trình tìm kiếm thủ phạm thực sự - thứ đang thực sự cản trở ngân hàng đạt được mục tiêu của mình - chúng tôi cần phải bắt đầu từ đâu đó.

Một nơi hợp lý để xem xét là những điểm chung giữa các PBI thất bại (phần lớn trong số đó).

Chúng tôi bắt đầu bằng cách loại bỏ những thứ mà chúng tôi có thể thấy không phải là điểm chung vì chúng thay đổi rất nhiều:

- Phần nào của hệ thống: một số PBI chỉ ảnh hưởng đến backend, số khác chỉ ảnh hưởng frontend, số khác nữa ảnh hưởng cả hai phần
- Đội nào thực hiện công việc—có vẻ như không quan trọng ai làm công việc đó, xác suất thất bại vẫn rất cao
- Product Owner nào viết công việc—cũng giống như với các đội

Sau đó chúng tôi bắt đầu xem xét những điểm chung. Danh sách không quá dài, nhưng cũng không ngắn:

- Các đội kỹ thuật
- Các Product Owner
- Ban lãnh đạo
- Văn hóa
- Môi trường phát triển
- Quy trình kỹ thuật
- Kỹ thuật viết yêu cầu
- Lĩnh vực (tài chính)
- Các dịch vụ phụ thuộc

Nhiều yếu tố trong số đó cũng có thể loại bỏ ngay lập tức. Các đội, Product Owner, ban lãnh đạo, văn hóa và môi trường phát triển đều đã được cải thiện gần đây mà không có tác động thực sự đến đầu ra có ý nghĩa. Chúng tôi đã trực tiếp giúp cải thiện quy trình kỹ thuật và cách viết yêu cầu, đồng thời xác nhận rằng những cải tiến đó đã được duy trì, nhưng vẫn không giúp ích được gì.

Khó có thể đổ lỗi cho lĩnh vực. Tài chính là một trong những loại tính toán lâu đời nhất trong lịch sử được ghi nhận. Nó đã rất trưởng thành. Hơn nữa, các ngân hàng khác vẫn đang triển khai phần mềm, điều này đã chứng minh giả thuyết (rõ ràng là khiên cưỡng) rằng các ngân hàng đơn giản là không thể làm được điều đó là sai.

Các dịch vụ phụ thuộc cũng không thể bị đổ lỗi, vì chúng cũng gặp nhiều khó khăn trong việc thay đổi như sáng kiến mà chúng tôi đang xem xét...

...nhưng điều đó khiến chúng tôi suy nghĩ: Nếu chúng tôi bắt đầu phân tích các nguyên nhân thất bại thì sao?

1.8: Mổ xẻ mầm mống của thất bại

Một PBI thất bại trong việc cải thiện sản phẩm bởi vì đội đã làm, như họ vẫn hay làm, một thứ gì đó hoàn toàn ngẫu nhiên và gần như không liên quan gì đến yêu cầu. Điều đó, rõ ràng, là dấu hiệu cho thấy họ không hiểu hạng mục công việc. Vì vậy sự hiểu biết là một ứng viên lớn, mặc dù chúng tôi đã phần nào làm việc về vấn đề đó khi giúp họ áp dụng BDD.

Một PBI khác thất bại vì họ tính toán sai. Đó là một bằng chứng khác cho thấy sự hiểu biết có thể là vấn đề cốt lõi.

Một hạng mục công việc thứ ba mà chúng tôi phân tích không thực sự được Product Owner kiểm soát đúng mức—cô ấy chỉ đơn giản phê duyệt khi đội nói đã đến lúc đóng. Điều đó có phần làm lung lay giả thuyết của chúng tôi, nhưng vẫn có thể lập luận rằng cô ấy không hiểu hạng mục công việc này phù hợp như thế nào với kế hoạch cấp cao hơn.

Có lẽ vậy. Phần nào đó. Nếu chúng tôi nhìn nhận theo cách đó một cách kỹ lưỡng.

Sau đó chúng tôi tình cờ gặp một PBI hoàn toàn không phù hợp với mô hình này. Đội có vẻ như đã hiểu—mặc dù không có cách nào để xác minh liệu họ có thực sự hiểu hay không. Nhưng điều đó không quan trọng: họ chưa bao giờ có cơ hội để thành công hay thất bại bởi vì họ gặp phải một phụ thuộc cần được cập nhật và phải hoãn công việc của họ đến vài Sprint sau.

Ngay cả khi họ *không* hiểu những gì họ được giao làm, họ cũng chưa bao giờ có cơ hội với hạng mục backlog đó, do đó sự hiểu biết **không phải** là vấn đề trong trường hợp đó.

Tất nhiên, một ngoại lệ không phải là bằng chứng bác bỏ một nguyên nhân gốc rễ cụ thể, nhưng nó đã kích thích sự tò mò của chúng tôi. Chúng tôi bắt đầu tìm kiếm các bằng chứng phản bác khác.

Và chúng tôi đã tìm thấy. Có những hạng mục công việc:

- Thất bại vì đội biết họ không hiểu, nhưng không thể tìm được chuyên gia về nghiệp vụ để giải quyết vấn đề
- Thay đổi thành trải nghiệm tệ hơn cho người dùng như một giải pháp tạm thời để phù hợp với cách hoạt động của các dịch vụ upstream
- Phải hoãn lại vì các phụ thuộc upstream chưa sẵn sàng
- Không thể hoàn thành vì người kiểm thử không thể thu thập dữ liệu kiểm thử kịp thời
- Đã đóng nhưng phải làm lại vì bản thân yêu cầu đã sai
- Thất bại vì đội không nhận ra mức độ phức tạp của code hiện có
- Đơn giản là không được ước lượng
- Bị ước lượng thấp một cách nghiêm trọng³
- Thay đổi giữa Sprint vì Product Owner cuối cùng đã có được kiến thức về nghiệp vụ cần thiết
- Có vẻ như thay đổi sau Sprint (từ góc nhìn của đội) vì Product Owner và đội chưa bao giờ thống nhất về ý nghĩa của nó

Danh sách còn dài, nhưng vậy là đủ cho câu chuyện này.

Có thể lập luận rằng mỗi vấn đề trên đều liên quan đến “sự hiểu biết” theo một cách nào đó—và chắc chắn là thiếu hiểu biết đã *liên quan* đến nhiều trường hợp—nhưng điều đó không có nghĩa là thiếu hiểu biết là nguyên nhân chính... đặc biệt khi chúng tôi đã làm một số công việc về hiểu biết chung và điều đó không thực sự giúp ích được gì.

Rồi chúng tôi chợt nhận ra. Có một yếu tố cơ bản hơn còn thiếu. Trong những trường hợp liên quan đến sự hiểu biết kém, đó chỉ là nguyên nhân trực tiếp.

Nguyên nhân gốc rễ còn rộng hơn nhiều.

1.9: Điểm Giao Nhau

Điều duy nhất mà mọi PBI chúng tôi phân tích về nguyên nhân thất bại đều có chung là: Tất cả đều được **bắt đầu quá sớm**.

Khi một hạng mục công việc thất bại vì nhóm biết họ không hiểu vấn đề và không thể tìm được chuyên gia để giúp họ hiểu, điều đó có nghĩa là nhóm đã bắt đầu một PBI trong khi biết rằng họ không hiểu vấn đề.

³Chúng tôi không chỉ nói rằng họ đã ước tính sai. Có vẻ như các nhóm chỉ đơn giản là điền giá trị “3” vào tất cả các trường ước tính cho một loạt PBI.

Khi một hạng mục backlog phải thay đổi thành trải nghiệm tệ hơn do cách thức hoạt động của một dịch vụ upstream, điều đó có nghĩa là công việc đã được bắt đầu mà không thực sự hiểu về tác động của dịch vụ upstream đó.

Việc trì hoãn do một dependency upstream không sẵn sàng vào lúc kết thúc có nghĩa là không có gì đảm bảo về sự sẵn sàng của nó ngay từ đầu.

...và tất cả các trường hợp khác cũng vậy: Người kiểm thử không có dữ liệu sẵn sàng hoặc không biết cách lấy dữ liệu trước khi PBI bắt đầu, yêu cầu chưa được xem xét kỹ trước khi giao cho nhóm, code chưa được kiểm tra trước khi công việc được cam kết, ước tính không đầy đủ hoặc không được thực hiện, thiếu kiến thức về lĩnh vực, và tất nhiên, hiểu biết chung chưa được xác minh.

Vấn đề, hóa ra là, việc triển khai đã bắt đầu trên các hạng mục công việc trước khi những hạng mục đó thực sự *sẵn sàng*.



Theo kinh nghiệm của chúng tôi, hầu hết các hạng mục công việc thất bại trong việc hoàn thành là do chúng **chưa sẵn sàng** khi bắt đầu triển khai.

Vì vậy chúng tôi bắt đầu giúp The Bank với vấn đề đó.

Ở thời điểm này, bạn có thể nghĩ rằng chỉ cần nói rằng PBI cần phải sẵn sàng là đủ tốt rồi. Tuy nhiên, hóa ra việc thực hiện không dễ dàng như vậy. “Mua thấp, bán cao” cũng là một ý tưởng đơn giản tương tự.

Còn thiếu một mảnh ghép cần thiết để đưa lời khuyên tốt vào thực tiễn.

1.10: Bước Ngoặt Lớn

Chúng tôi đã tìm thấy mảnh ghép còn thiếu của câu đố đã giải phóng họ, và từ đó, giải phóng cả sáng kiến.

Khi chúng tôi hoàn thành dự án đó, các kỹ sư vẫn còn kém hơn nhiều so với mức trung bình về kỹ năng. Các PO vẫn chưa có đủ kỹ năng cần thiết. Văn hóa vẫn chưa được cải thiện...

Tuy nhiên sản phẩm cuối cùng đã bắt đầu tiến triển và cuối cùng đã được ra mắt.

Đến cuối cuốn sách này, bạn sẽ biết mảnh ghép còn thiếu là gì, và những gì cần thiết để đưa nó vào vị trí. Và bạn sẽ có khả năng giải phóng các tổ chức dường như đang bị kìm hãm bởi một bức tường vô hình.

Ghi Chú Nhanh về Phạm Vi

Cuốn sách này nói về một vấn đề rất cụ thể và phổ biến: sự thiếu cấu trúc, rõ ràng và trưởng thành trong quá trình bàn giao giữa kinh doanh và kỹ thuật. Nó giả định rằng đã có thứ gì đó được chọn để triển khai, và tập trung vào việc đảm bảo rằng công việc xây dựng diễn ra với sự hiểu biết chung, sự sẵn sàng, và khả năng theo dõi hoàn thành.

Các kỹ thuật trong cuốn sách này không nói cho bạn biết nên xây dựng cái gì, tại sao xây dựng nó, hoặc làm thế nào để biết liệu đó có phải là thứ đúng đắn để xây dựng hay không. Nếu tổ chức của bạn thiếu quản lý sản phẩm thực sự hoặc vòng phản hồi có ý nghĩa, chúng tôi không cố gắng giải quyết điều đó ở đây. Thay vào đó, chúng tôi đề xuất một cách để làm cho những khoảng trống đó trở nên rõ ràng hơn, và để giảm chi phí của việc phát hiện ra rằng bạn đã sai.

Khi được sử dụng trong bối cảnh phù hợp, giải pháp này mang lại luồng công việc, sự an toàn và sự rõ ràng. Nhưng giống như bất kỳ hệ thống nào, nó có thể bị áp dụng sai—đặc biệt là khi được sử dụng một cách riêng lẻ hoặc không có nhận thức đầy đủ.

Chương 2: Chi phí của việc thiếu nền tảng

Sẽ là hợp lý khi dành một chút thời gian để tìm hiểu chính xác vấn đề này có thể *tệ* đến mức nào đối với một số tổ chức.

Chúng tôi nhận thấy có ba “nhóm” vấn đề chính mà sự thiếu sẵn sàng tạo ra:

- Thiếu hoặc không đầy đủ sự hiểu biết chung giữa và trong nội bộ Product và Engineering
- Thiếu kiểm soát về mục tiêu thực sự của một PBI và khi nào nó thực sự hoàn thành
- Thiếu kiểm soát về thời điểm một hạng mục công việc có thể bắt đầu giai đoạn thực hiện trong vòng đời của nó

Ngoài ra, chúng tôi nhận thấy việc thay đổi những điều này có thể khá thách thức. Điều đó là hợp lý: thay đổi luôn khó khăn.

Thói quen cũ khó bỏ và thói quen mới khó hình thành. Theo kinh nghiệm tư vấn của chúng tôi, chúng tôi nhận thấy mọi người *rất* dễ quay lại thói quen cũ và tương đối khó để họ hình thành thói quen mới.

Vì vậy, bạn phải có một cơ chế để thực thi thói quen mới và ngăn cản thói quen cũ.

Để giải quyết điều này, chúng tôi tin rằng còn thiếu một thành phần khác về trách nhiệm giải trình và khả năng truy vết.

2.1: Bức tranh ghép không có hình mẫu

Bạn đã bao giờ thử ghép một bức tranh ghép mà không có hình mẫu trên hộp chưa? Bạn có thể làm được, nhưng sẽ chậm hơn, gây nhiều bức bối hơn, và đầy những khởi đầu sai lầm.

Bạn đạt được tiến độ, rồi lại phải tháo ra. Bạn đắn đo về việc mảnh nào khớp với mảnh nào. Bạn nghĩ mình đang làm cùng một bức tranh, cho đến khi nhận ra là không phải.

Đó chính là cảm giác thường thấy trong phát triển phần mềm.

Product backlog đầy ắp. Sprint đang chạy. Mọi người đều làm việc chăm chỉ.

Nhưng khi không có một bức tranh chung về những gì chúng ta đang xây dựng, sự đồng bộ trở thành may mắn, không phải một hệ thống.

Không có sự rõ ràng, ngay cả những đội ngũ giỏi nhất cũng cảm thấy thất vọng, kiệt sức, và cảm giác nỗ lực của họ không được đánh giá đúng mức.

2.2: Một câu châm ngôn cũ và một thực tế phũ phàng

Có lý do tại sao rất nhiều phương pháp về viết yêu cầu phần mềm, thậm chí một số thực hành kỹ thuật, tập trung nhiều vào việc tạo ra sự hiểu biết chung giữa người yêu cầu và đội ngũ thực hiện. Không có gì gây ra hỗn loạn bằng việc các kỹ sư không thực sự biết họ đang phải xây dựng cái gì.



“Đầu vào rác, đầu ra rác” là một châm ngôn, không phải một câu nói sáo rỗng.

Tiếng Anh chứa đầy những mâu thuẫn...

- Bạn có thể sanction hành động của ai đó. Có thể điều đó có nghĩa là bạn đã cho phép họ trước đó hoặc có thể là bạn đang lên án họ sau sự việc.
- Bạn có thể “lightly dust” (phủ nhẹ) thứ gì đó, nhưng nếu đó là một chiếc tủ, điều đó có nghĩa là bạn đang lau bụi khỏi nó trong khi nếu nói về một chiếc bánh beignet, bạn đang rắc bụi lên nó.
- Nếu bạn “hold up” một đội, bạn có thể là lý do giúp đội đó tiếp tục hoạt động hoặc có thể là lý do khiến họ không thể tiến lên được.

Từ tự đối lập có thể là những ví dụ nổi bật nhất, nhưng chúng chỉ là một loại sự mơ hồ. Một số từ không chỉ là đối lập với chính nó, mà còn có nhiều nghĩa thay thế có thể gây nhầm lẫn khác.

“In this clip, he clipped a coupon from a news paper and clipped it to the paper on his clipboard along with the other clippings while a clipper in the background was moving along at a decent clip.”

Không chỉ tiếng Anh, mà *tất cả* ngôn ngữ tự nhiên mà chúng tôi biết đều có đặc điểm này.

Và, tuy vậy, đó là những công cụ duy nhất chúng ta có để làm việc khi xác định yêu cầu.

Kết quả là, khi một đội ngũ kỹ thuật chưa *xác nhận* rằng hiểu biết của họ về một yêu cầu giống với hiểu biết của người yêu cầu, đội ngũ đó đang phụ thuộc vào may mắn. Nghĩa là, kết quả tốt nhất có thể là họ chọn đúng cách hiểu và người yêu cầu không thay đổi ý định trong quá trình thực hiện.

Kết quả đó hoàn toàn không được đảm bảo.

2.3: Một số kết quả phổ biến

Không có sự hiểu biết chung đã được xác nhận, các đội phải đối mặt với nhiều rủi ro.

Nhìn chung, kết quả gây đau đớn phổ biến nhất là đội ngũ đơn giản là xây dựng sai thứ cần làm.

Thời điểm họ phát hiện ra điều đó có thể được kiểm soát bởi các thuộc tính khác nhau trong quy trình của họ. Ví dụ, một quy trình Scrum lành mạnh có thể phát hiện ra sự hiểu lầm kiểu đó rất sớm trong quá trình thực hiện trong khi một quy trình Waterfall có khả năng rất cao trì hoãn việc phát hiện như vậy đến hàng tháng.



Định nghĩa: Waterfall

Một mô hình phát triển phần mềm tuần tự trở nên phổ biến vào cuối thế kỷ 20. Nó được mô tả lần đầu trong [bài báo năm 1970 của Winston W. Royce](#), minh họa việc phát triển như một chuỗi các bước xếp tầng—yêu cầu, thiết kế, thực hiện, kiểm thử, và tiếp tục—mỗi bước đổ vào bước tiếp theo như một thác nước. Mặc dù Royce trình bày mô hình này như một ví dụ về những gì *không nên* làm, nhưng ngành công nghiệp đã áp dụng nó như một bản thiết kế cho phát triển quy mô lớn.

Waterfall cũng nổi tiếng với việc nhóm các công việc tương tự vào các giai đoạn lớn, liên tiếp—một đặc điểm được gọi là **phát triển theo lô lớn**. Việc phân lô này gần như đảm bảo **học hỏi muộn**: các đội không nhận được phản hồi về các quyết định trước đó cho đến rất muộn trong quy trình. Lỗi được phát hiện muộn tốn kém hơn để sửa chữa. Những người thực hành Agile phê bình Waterfall vì lý do này, ưa thích các chu kỳ nhỏ hơn, lặp đi lặp lại cho phép phát hiện sớm hơn và điều chỉnh hướng đi.

Tuy nhiên, đến một lúc nào đó, một nhóm làm việc với sự hiểu biết “sai” (thực ra là khác) về một yêu cầu sẽ phải đối mặt với yêu cầu “đúng” (cũng thực ra là khác). Một lần nữa, mức độ lãnh mạnh của tổ chức sẽ ảnh hưởng đến hình thức đối mặt đó và tác động của nó, nhưng điều này gần như luôn xảy ra.

Trong hầu hết các trường hợp, điều này dẫn đến một số hình thức làm lại. Người yêu cầu (thường là Quản lý Sản phẩm) sẽ phải yêu cầu thay đổi để chuyển từ những gì các nhóm thực hiện đã xây dựng sang những gì anh ta thực sự muốn.

Một biểu hiện rất phổ biến khác là người yêu cầu tiếp tục buộc nhóm phải chịu trách nhiệm theo hiểu biết ban đầu của anh ta về những gì anh ta đã yêu cầu.

Các nhóm có thể dễ dàng diễn giải điều này như việc Quản lý Sản phẩm thay đổi ý kiến. Tệ hơn, nó thực sự có thể *khuyến khích* các bên liên quan hình thành thói quen thay đổi ý kiến—trì hoãn các hạng mục công việc cho đến khi mọi thứ *hoàn hảo*, làm kiệt sức các nhóm, và che mắt các nhà quản lý cấp cao về tiến độ.

2.4: Khi nào một Câu đố Được Coi là Hoàn thành?

Quay trở lại với ẩn dụ về việc lắp ráp câu đố, hãy suy nghĩ về câu hỏi này: Hoàn thành một câu đố nghĩa là gì?

Một người lắp ráp câu đố ngây thơ, như Max, sẽ đơn giản nói rằng “tất cả các mảnh ghép được gắn với các mảnh lân cận đúng vị trí với hình ảnh hướng lên trên.”

Tuy nhiên, một người lắp ráp câu đố sành sỏi, như Luniel, biết rằng còn nhiều điều hơn thế nữa.

Có thể bạn chỉ đang lắp ráp câu đố để giải trí. Bạn sẽ hoàn thành nó, ngắm nhìn một lúc, rồi tháo ra và cho vào hộp.

Mặt khác, có thể bạn muốn đóng khung và treo nó lên tường. Nếu vậy, có những việc bổ sung cần phải làm:

1. Đặt nó lên bảng nghệ thuật
2. Vận chuyển đến thợ đóng khung
3. Chờ hoàn thành việc đóng khung
4. Vận chuyển về nơi treo
5. Treo lên tường hoặc trưng bày theo cách khác

Việc hiểu rằng đây là một phần của công việc là cần thiết để hoàn thành đúng cách việc lắp ráp câu đố. Lý do hiển nhiên là để bạn biết có bao nhiêu công việc cần làm. Thực hiện tất cả các bước bổ sung đó tốn công hơn nhiều so với việc chỉ tháo ra và cất đi.

Vấn đề còn sâu xa hơn thế. Hãy tưởng tượng tình huống này...

Bạn đã hoàn thành câu đố với ý định đóng khung, bạn để nguyên nó tại chỗ, nhưng bạn quên không nói với người khác trong nhà rằng bạn định đóng khung nó. Người đó đi ngang qua và thấy câu đố đã hoàn thành ở một không gian mà họ cần dùng. Vì vậy họ tháo nó ra và cho vào hộp, biến chiến thắng thành thất bại trong quá trình đó.

Còn có một lý do tinh tế hơn nữa: Cách bạn dự định hoàn thành câu đố ảnh hưởng đến những bước bạn muốn thực hiện từ sớm trong quá trình. Ví dụ, bạn cần làm một tấm biển nhỏ ghi “Vui lòng không tháo rời!”



Cũng đáng lưu ý rằng bạn có thể muốn có một tấm biển trong trường hợp bạn đang lắp ráp câu đố để giải trí để đảm bảo người khác không can thiệp bằng cách hoàn thành câu đố thay bạn.

Bạn cũng cần đảm bảo rằng bạn đang lắp ráp câu đố trên bề mặt phù hợp. Nếu bạn lắp ráp một câu đố một nghìn mảnh trên bàn cà phê mặt kính rồi cố gắng chuyển nó sang bảng nghệ thuật, việc di chuyển sẽ rủi ro và tốn công sức hơn nhiều so với việc lắp ráp trực tiếp trên bảng nghệ thuật ngay từ đầu.

Điều này tương đồng rất tốt với phát triển phần mềm.

Bạn cần thực sự biết “hoàn thành” nghĩa là gì để không bị bất ngờ về lượng công việc cần làm, không có bất đồng về nó ở cuối cùng, và bạn có thể thực hiện các bước chuẩn bị cần thiết để đảm bảo hoàn thành một hạng mục công việc một cách suôn sẻ và hiệu quả.

2.5: Tác động đến các Nhóm

Nếu bạn không có hiểu biết đủ chặt chẽ về việc hoàn thành là gì đối với một hạng mục công việc cụ thể, bạn sẽ gặp phải một số rủi ro.



Chúng ta sử dụng từ “rủi ro” một cách tương đối ở đây, vì chúng gần như là những điều chắc chắn xảy ra.

Các nhóm kỹ thuật trong tình huống này thường phát hiện ra rằng họ thậm chí không có sự đồng thuận nội bộ về việc hoàn thành một hạng mục công việc trông như thế nào. Không

hiếm khi các lập trình viên và người kiểm thử thấy mình đang tranh luận về ý nghĩa thực sự của một yêu cầu khi đã đi được nửa Sprint. Thậm chí hai lập trình viên hoặc hai người kiểm thử cũng có thể gặp phải những bất đồng tương tự.

Hơn nữa, các đội phát triển thường tập trung vào công việc họ làm nhiều nhất (lập trình và kiểm thử). Điều này có nghĩa là họ dễ quên các loại công việc khác họ cần làm, chẳng hạn như tài liệu, đánh giá bên ngoài, đào tạo các nhóm khác (ví dụ nhóm hỗ trợ), các bước chuẩn bị để hỗ trợ triển khai hoặc phát hành, và phê duyệt từ các phòng ban khác.

Cuối cùng khi mọi người nhận ra rằng công việc “phụ thêm” này cần phải được thực hiện, họ hoàn toàn bất ngờ—thường thì họ phải dừng bất cứ việc gì đang làm và chuyển ngữ cảnh để quay lại hoàn thành công việc mà họ nghĩ đã hoàn tất.

Những người yêu cầu công việc có thể dễ dàng giữ công việc mở một cách không cần thiết. Đôi khi với ý định tốt nhất—như cố gắng giữ nhóm chịu trách nhiệm với yêu cầu “thực sự”. Những lúc khác, điều này xảy ra vì Product Owner (ví dụ) quen với việc có thể giữ một PBI mở theo ý muốn, nên họ dùng nó để nhét thêm chức năng vào một hạng mục vào phút cuối. Đôi khi họ thậm chí làm vậy vì họ đã thay đổi ý định về việc cần làm gì giữa chừng khi đang thực hiện.

Điều này có thể cực kỳ làm nản lòng đội ngũ kỹ thuật. Hầu hết các nhà phát triển và kiểm thử phần mềm đều muốn cảm thấy họ đang tiến bộ. Nếu họ liên tục bị nói rằng những gì họ làm là sai, họ có thể sẽ mất đi một phần nhiệt huyết.

Một số nhóm thậm chí còn không thêm kiểm tra xem những gì họ làm là đúng hay sai. Họ chỉ đóng hạng mục công việc và yêu cầu được “ghi nhận công” để họ có thể “đạt chỉ số tốt”.

2.6: Rủi Ro Của Việc Làm Quá Ít hoặc Quá Nhiều

Một rủi ro của việc không định nghĩa đúng đắn “hoàn thành” cho mỗi hạng mục công việc là tổ chức nghĩ rằng công việc đã hoàn thành khi chưa hoàn thành, hoặc không nhận ra công việc đã hoàn thành khi thực sự đã xong.

Kết quả tồi tệ nhất thường là việc sai lầm được đưa lên môi trường production và không ai biết điều đó đã xảy ra. Nếu nhóm có hiểu sai về ý nghĩa của “hoàn thành” và triển khai dựa trên sự hiểu sai đó, kết quả có thể là thảm họa.

Lỗi và sự không hài lòng của khách hàng đã đủ tệ, nhưng điều này còn có thể dẫn đến những vấn đề nghiêm trọng hơn:

- Mất mát hoặc hỏng dữ liệu

- Lỗi hỏng bảo mật
- Thời gian ngừng hoạt động hoặc mất quyền truy cập hệ thống
- Giảm thị phần
- Vi phạm quy định

Danh sách cứ tiếp tục và mỗi vấn đề tiềm ẩn lại tệ hơn vấn đề trước.

Đôi khi vấn đề là bạn chưa hoàn thành nhưng bạn nghĩ mình đã xong. Ngược lại cũng có thể nguy hiểm không kém. Khi các kỹ sư không biết đâu là đích đến, họ có xu hướng “mạ vàng” (thêm tính năng phụ). Họ có thể làm vậy để “làm cho tính năng đẹp hơn”, nhưng họ cũng có thể làm vậy vì họ hy vọng rằng việc tăng số lượng tính năng sẽ cho họ cơ hội cao hơn để đạt được mục tiêu.

Tất cả công việc phụ thêm này, cũng như công việc làm lại tương ứng tích tụ thành một lượng thời gian, công sức và tiền bạc lãng phí khổng lồ. Nó khiến ngày giao hàng bị trễ và làm tổn hại đến uy tín.

Hơn nữa, giờ đây, có một khả năng ngày càng tăng là một sai lầm thực sự dẫn đến cuộc nổi dậy kiểu *Terminator* của máy móc chống lại loài người. Hai mươi năm trước chúng ta thường viết về điều đó như một trò đùa. Giờ đây, nó là một khả năng xa xôi.

Thực tế, chúng tôi đã hỏi một trong những AI nổi bật nhất câu hỏi này, và đây là câu trả lời:

“AI đang phát triển nhanh hơn bất kỳ ai mong đợi, nhưng nó đang phát triển trên những hệ thống mong manh, những yêu cầu mơ hồ, và những tổ chức sản phẩm không thể truy xuất được lý do tại sao họ xây dựng những gì họ đã xây dựng. Đó không phải là vấn đề kỹ thuật; đó là vấn đề về sự rõ ràng. Càng nhiều nhiễu AI tạo ra, càng nguy hiểm khi di chuyển nhanh mà không có cấu trúc. Khi các nhóm xây dựng trên sương mù, tất cả những gì AI làm là khuếch đại mơ hồ hỗn loạn. Nhưng khi các nhóm xây dựng trên tín hiệu—trên sự hiểu biết chung, tính cụ thể về hành vi, và kiến thức sản phẩm được kiểm soát phiên bản—AI trở thành chất xúc tác thay vì một trở ngại.”

2.7: Bạn Ghép Hình Ở Đâu?

Hãy mở rộng phép ẩn dụ về ghép hình thêm một lần nữa.

Bạn có thể ghép hình ở bất cứ đâu không? Nếu bạn có một bộ ghép hình 4.000 mảnh khi ghép xong sẽ dài gần năm feet theo một chiều và hơn ba feet theo chiều kia, bạn không thể chỉ chọn ngẫu nhiên một chỗ và bắt đầu ghép. Không thể làm vậy mà không gặp những rắc rối nghiêm trọng trước khi hoàn thành.

Một bộ ghép hình lớn như vậy cần cả thời gian và không gian. Bạn phải phân bổ không gian và tìm cách đảm bảo trạng thái của bộ ghép hình được duy trì theo thời gian.

Nếu bạn bắt đầu ghép hình trên một chiếc bàn nhỏ quá hẹp, bạn sẽ không thể hoàn thành nó mà không phải chuyển nó sang vị trí khác. Việc di chuyển đó sẽ cực kỳ khó khăn do trạng thái mong manh của bộ ghép hình.

Nếu bạn chọn một vị trí ngẫu nhiên trong hành lang đủ rộng, mọi người sẽ hoặc là đi qua nó hoặc bị cản trở, vì vậy việc duy trì không thể được đảm bảo mà không ảnh hưởng đáng kể đến sinh hoạt trong nhà của bạn.

Nếu bạn bắt đầu làm việc trên một tấm bảng nghệ thuật, nhưng tấm bảng không đủ lớn, bạn sẽ có thể bảo toàn trạng thái của những gì bạn đã làm, nhưng bạn sẽ không thể hoàn thành câu đố mà không cần một số hình thức chuyển đổi nào đó.

Nếu câu đố đã từng bị trẻ nhỏ nhai trước đó, tốt nhất là nên đếm số mảnh ghép... bởi vì thà đếm đến 3999 một lần và nhận ra rằng bạn sẽ không bao giờ hoàn thành được còn hơn là đầu tư không biết bao lâu để gần như hoàn thành một câu đố mà bạn sẽ không bao giờ có thể hoàn thành.

Có cả một danh sách những việc cần làm trước khi bạn bắt đầu xây dựng câu đố của mình. Việc thực hiện những điều trong danh sách không đảm bảo thành công, nhưng việc *không* thực hiện chúng gần như chắc chắn dẫn đến thất bại hoặc những rắc rối nghiêm trọng.

Điều tương tự cũng đúng với việc phát triển phần mềm, nhưng với mức độ phức tạp cao hơn.

2.8: Khi Nào Bắt Đầu Triển Khai?

Về cơ bản, có thể khó xác định khi nào một PBI sẵn sàng để được triển khai nếu không có một định nghĩa tốt về điều đó.

Hãy suy nghĩ về điều này: Làm thế nào bạn *biết được*?

Bạn có xem xét mọi thứ lặp đi lặp lại cho đến khi bạn quyết định đã đến lúc?

Có ai đó quyết định theo cảm tính?

Nó có xảy ra tự động vào đầu một chu kỳ lặp?

Chúng tôi đã thấy nhiều đội đẩy công việc vào các Sprint mà chưa hề sẵn sàng để triển khai chỉ vì họ đang chịu áp lực thời hạn. Có một số ý tưởng phổ biến về Scrum và Agile nói chung khiến mọi người làm điều này:

- Bạn phải có tất cả yêu cầu cho Sprint N được phát triển trong Sprint $N-1$
- Bạn nên “cứ bắt đầu” và xử lý những gì hỏng hóc dọc đường

Đây thực sự là hình ảnh phản chiếu của vấn đề “Làm thế nào bạn biết khi nào thì xong?” [đã đề cập trước đó](#) và nó có những hệ quả tương tự. Mọi người có thể đợi quá lâu để bắt đầu vì họ không biết một hạng mục công việc đã sẵn sàng và họ có thể bắt đầu quá sớm vì họ không biết nó chưa sẵn sàng.

2.9: Một Đội A Không Có Sự Sẵn Sàng

Không hiểu những gì cần thiết để một hạng mục công việc sẵn sàng có nhiều tác động có hại.

Một cách hiển nhiên mà một phần công việc có thể không sẵn sàng là khi Definition of Done (DoD) không đầy đủ, thiếu sót, hoặc không tồn tại. Điều này dẫn đến tất cả những vấn đề mà chúng ta đã đề cập trước đó liên quan đến việc không có Definition of Done.

Tuy nhiên, đó không phải là *khía cạnh* duy nhất của sự sẵn sàng. Có nhiều nhu cầu khác cần được thỏa mãn trước khi bắt đầu triển khai: ước tính, đánh giá rủi ro, và thu thập dữ liệu kiểm thử chỉ là một vài ví dụ phổ biến.

Không biết và không thỏa mãn những nhu cầu đó, một hạng mục công việc có thể tồn kém nhiều hơn mức cần thiết. Hãy xem xét một đội (Đội A) phụ thuộc vào một API đang được phát triển bởi một đội khác (Đội Khác). Nếu Đội A đưa ra nhiều giả định về cách API của Đội Khác sẽ hoạt động và viết mã dựa trên những giả định đó, có thể sẽ có nhiều việc phải làm lại khi họ phát hiện ra rằng Đội Khác đang làm việc theo một thực tế không khớp với giả định của Đội A. Nói cách khác, Đội A đã đánh cược và thất bại.

Tất cả công việc làm lại đó bắt nguồn từ việc API chưa sẵn sàng để được sử dụng bởi Đội A.

Đôi khi một phụ thuộc chưa được thỏa mãn không tạo ra công việc làm lại, nhưng ngay cả trong những trường hợp đó, nó vẫn có thể gây ra sự chậm trễ. Hãy tưởng tượng nếu Đội A và Đội Khác đã thống nhất về cách API nên hoạt động và mọi thứ diễn ra như kế hoạch, nhưng Đội Khác đơn giản là mất nhiều thời gian hơn dự kiến. Kết quả là, Đội A đơn giản là

không thể kiểm thử công việc của họ một cách thích hợp vào thời điểm nó được cho là hoàn thành và họ phải lùi thời hạn của mình.

2.10: Thất Bại Trong Việc Chú Ý Đến Lịch Trình và Khả Năng Sẵn Có của Nguồn Lực

Đôi khi các vấn đề có thể đơn giản như lịch trình hoặc nguồn lực. Một số hạng mục công việc cần những thành viên cụ thể trong đội. Nếu thành viên đó sắp đi nghỉ trong vài ngày tới, có lẽ đây không phải là thời điểm thích hợp để bắt đầu PBI mà không thể hoàn thành nếu thiếu sự tham gia của họ.

Chúng tôi thường xuyên nghe mọi người nói rằng không nên như vậy, nhưng nó thường xảy ra, bất kể thế nào. “Nhân sự có thể thay thế cho nhau” chỉ là một ảo tưởng viễn vông.

Điều tương tự cũng có thể nói về các tài nguyên phi nhân lực. Nếu bạn sẽ cần tài nguyên máy chủ để thực hiện kiểm thử tải, có lẽ bạn nên đảm bảo những tài nguyên đó thực sự sẽ có sẵn trước khi bạn bắt đầu hạng mục công việc kiểm thử tải. Nếu không, trong trường hợp tốt nhất bạn sẽ gặp phải sự chậm trễ đáng kể và có thể sẽ gây xáo trộn cho các nhóm/nhân viên khác trong khi bạn đang cố gắng vội vàng cung cấp những gì bạn cần.

Một hình thức thất bại khác từ việc khởi đầu sai lầm là khi một nhóm không có đủ kỹ năng cần thiết để hoàn thành công việc. Đôi khi đó là vấn đề nội bộ—chẳng hạn như một thành viên trong nhóm cần được đào tạo về một hệ thống mới hoặc nghiên cứu về một API mới. Những lúc khác, đó là vấn đề về lịch trình, chẳng hạn như khi bạn cần mượn một chuyên gia UX hoặc cơ sở dữ liệu từ nhóm nhân sự có kỹ năng. Thậm chí có thể là vấn đề tuyển dụng khi nhóm cần một chuyên gia và không thể hoàn thành hiệu quả một số loại công việc nhất định nếu thiếu người này.

2.11: Tác động từ Các Loại Khởi Đầu Sai Lầm Khác

Chúng tôi đã thấy các nhóm cam kết hoàn thành các hạng mục công việc trong các Sprint và thực hiện việc lập trình tương đối nhanh nhưng vẫn không thể hoàn thành việc kiểm thử. Điều này, tự nó có thể không gây ngạc nhiên, nhưng lý do thì khá bất thường: nhóm kiểm

thứ có thứ họ cần (như dữ liệu kiểm thử) mà họ chưa thu thập được trước khi Sprint bắt đầu và việc thu thập những dữ liệu đó hóa ra khó khăn hoặc tốn thời gian hơn họ dự đoán.

Kết quả là, các hạng mục công việc phải được chuyển sang Sprint tiếp theo đơn giản vì nhóm đã không đảm bảo họ thực sự sẵn sàng để hoàn thành nó trong thời gian được phân bổ trước khi họ bắt đầu.

Các nhóm đôi khi bắt đầu thực hiện các hạng mục công việc khi họ vẫn còn những câu hỏi chưa được giải đáp. Thực tế, nhiều người dường như nghĩ rằng làm như vậy khiến họ trở nên “Agile hơn”.

Điều này có thể gây ra một lượng lớn công việc phải làm lại, những bất ngờ, hoặc sự chậm trễ. Nếu câu trả lời cho câu hỏi còn bỏ ngỏ vi phạm một giả định đã được đặt ra, tất cả công việc dựa trên giả định đó phải được xem xét lại. Nếu câu hỏi còn bỏ ngỏ không được trả lời vào thời điểm hạng mục đó cần được đóng lại, thì hạng mục đó hoặc phải được đóng lại khi có thể chưa hoàn thành, hoặc phải giữ mở cho đến khi câu hỏi được trả lời.

Có thể là nhóm có một phụ thuộc nội bộ—một lỗi cần được sửa, một nhiệm vụ tiền đề phải được hoàn thành, *vân vân*. Nếu điều đó không được theo dõi đúng cách, nó có thể gây ra tất cả các vấn đề tương tự như một phụ thuộc bên ngoài chưa được thỏa mãn với thêm cảm dỗ chuyển đổi ngữ cảnh và sửa chữa nó.

2.12: Chi Phí Tích Lũy

Tất nhiên, những vấn đề như vậy tạo ra sự chậm trễ, làm lại công việc, và những kỳ vọng bị phá vỡ, nhưng mặt trái không dừng lại ở đó.

Bên cạnh sự lãng phí từ việc làm lại, điều này thường khiến các dự án bị trì hoãn. Nếu các nhóm đang cố gắng một cách điên cuồng để đóng các hạng mục tồn đọng và không bao giờ thực sự rõ ràng về việc cần làm gì để tạo ra một số tiến bộ thực sự, những thứ thực sự *cần* được hoàn thành thường bị bỏ qua.

Thường xuyên, mặc dù không phải lúc nào cũng vậy, điều này dẫn đến áp lực tăng cao để chuyển giao. Khi các dự án ngày càng bị chậm tiến độ, ban quản lý cấp cao có thể cố gắng đưa nó trở lại đúng hướng bằng cách yêu cầu mọi người làm việc nhanh hơn. Điều đó luôn luôn dẫn đến việc làm việc nhiều giờ hơn.

Điều này, đến lượt nó, có xu hướng làm xói mòn lòng tin và làm xấu đi văn hóa của một tổ chức. Các mối quan hệ đáng lẽ phải mang tính hợp tác lại trở nên đối đầu. Những người đáng lẽ phải làm việc cùng nhau để tìm ra giải pháp tốt nhất, nhanh nhất, lại chuyển hướng

năng lượng để chứng minh rằng, khi mọi thứ không thể tránh khỏi việc đi sai hướng, đó không phải là lỗi của họ.

Trong cuộc đua điên cuồng để có các tính năng và đóng các hạng mục công việc, các nhóm thường thấy mình đang cắt giảm các góc cạnh. Điều đó thực sự có nghĩa là họ đang để chất lượng (đặc biệt là chất lượng mã nguồn) bị ảnh hưởng. Điều đó, đến lượt nó, có nghĩa là họ đang đánh đổi năng suất trong tương lai cho ảo tưởng về tiến độ ở hiện tại.

Khi điều kiện làm việc ngày càng trở nên khó chịu, những nhân tài chủ chốt bắt đầu không còn gắn kết hoặc thậm chí bắt đầu tìm kiếm cơ hội ở nơi khác.

Các tổ chức có hành vi như vậy đang “ăn hết hạt giống”, như người ta vẫn nói, theo nhiều cách. Nền tảng mã nguồn trở nên khó bảo trì hơn và những người lẽ ra sẽ bảo trì nó đều bị đẩy đi.

Nếu có mặt tích cực nào, chúng tôi cũng không thể nhìn thấy.

Chương 3: Giới thiệu Requirements Maturation Flow (RMF)

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.1: RMF Không Phải Là Gì

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.2: RMF Là Gì

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.3: Việc Áp dụng Từng phần được Hỗ trợ và Khuyến nghị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.4: RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.5: RMF 2

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

3.6: RMF 3

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 4: Liệu Có Phải Là Agile?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

4.1: “Con người và Sự tương tác”, “Phần mềm chạy được”

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

4.2: Hợp tác với Khách hàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

4.3: Đáp ứng với Thay đổi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

4.4: Tính minh bạch

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

4.5: Phù hợp với Quy trình, Nhất quán với Agile

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phần II: Tạo Điều Kiện cho Sự Sẵn Sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 5: Phần Mở Rộng Đầu Tiên

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

5.1: Công Việc Chuẩn Bị là *Công Việc*

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

5.2: Tự Nhiên Hóa Công Việc Chuẩn Bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

5.3: Một Sự Cố Minh Họa

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

5.4: Tác Động Qua Lại

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

5.5: Chức năng của RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 6: Tại Sao Mọi Người Không Làm Điều Này?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.1: Công Việc Chuẩn Bị Như Một Công Dân Hạng Hai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.2: Sự Dị Ứng với Công Việc Không Sản Xuất

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.3: Vì Vậy Nó Bị Chôn Vùi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.4: Ảnh Hưởng của Quản Lý Dự Án

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.5: Khuôn Mẫu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.6: Dự Án và Ước Tính

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.7: Cách Những Ước Tính Không Phải Ước Tính Ảnh Hưởng đến Công Việc Chuẩn Bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.8: Đo Lường Tốc Độ, Không Phải Vận Tốc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.9: Đo Lường Sai, Kết Quả Sai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

6.10: Đâu Không Phải Là Nguồn Gốc Của Vấn Đề

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 7: Công việc Sẵn sàng Rõ ràng (RMF 1)

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.1: Tích hợp với Synapse Framework™

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.2: Cấu trúc của RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.3: Hành vi: Dự trữ Năng lực cho Hợp tác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.4: Tạo tác phẩm: Hạng mục công việc chuẩn bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.5: Hoạt động: Cuộc họp Cộng tác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.6: Hành vi: Tiếp tục Hợp tác Cho đến khi Đạt được Sự Hiểu biết Chung

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.7: Hành vi: Luôn Xác nhận Sự hiểu biết chung

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

7.8: Cách RMF 1 Thay đổi Quy trình làm việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 8: Các tác động của RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

8.1: Cuộc sống trước RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

8.2: Trước: Thời gian dành cho việc Tìm hiểu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

8.3: Sau đó: Thời gian dành cho việc thấu hiểu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

8.4: Cuộc sống sau khi áp dụng RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

8.5: Nền tảng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 9: Đưa RMF 1 Vào Thực Tiễn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.1: Giáo dục

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.2: Yêu Cầu Tối Thiểu theo Loại Nhóm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.3: Thỏa Thuận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.4: Chuẩn Bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.5: Thử nghiệm thí điểm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.6: Triển khai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.7: Theo Dõi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.8: Tuyên Bố Thành Công

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.9: Duy Trì Cảnh Giác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.10: Còn về phần “Làm thế nào”?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

9.11: Đã Đến Lúc Biến Nó Thành Hiện Thực!

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phần III: Kiểm soát Hoàn thành Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 10: Nhu Cầu Tiếp Theo

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.1: Khoảng Trống Diễn Giải

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.2: Thu Hẹp Khoảng Trống Diễn Giải

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.3: Một Lựa Chọn Thứ Ba: Không “Khoảng Trống Dao Động”

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.4: Tác Động Tiềm Ẩn đến Việc Hoàn Thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.5: Tác động Tiềm ẩn đến Việc Thực thi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.6: Giải pháp Đề xuất: Không Để Chỗ cho Hiểu lầm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.7: Lợi ích

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.8: Về Nỗi lo Tê liệt Phân tích

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

10.9: Nhu Cầu Tiếp Theo: Định Nghĩa Hoàn Thành Tùy Chỉnh

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 11: Mọi Người Thường Làm Gì

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.1: Nếu Nó Tuyệt Vời Như Vậy, Tại Sao Mọi Người Không Làm?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.2: Con Đường Từ Đại Học Đến Huấn Luyện Viên

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.3: Quá Tải Huấn Luyện Viên

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.4: Một Cách Mọi Người Làm DoD: Không Làm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.5: Chỉ Có Tiêu Chí Chấp Nhận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.6: Định nghĩa Hoàn thành Toàn cục

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.7: Không Có Răng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

11.8: Tổng kết: Thuật ngữ “DoD” Phổ biến hơn Định nghĩa Hoàn thành Thực tế

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 12: Xác định Định nghĩa Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.1: Về Một Hạng mục Công việc Duy nhất

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.2: Tính hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.3: Tính chính xác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.4: Cấu trúc của DoD

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.5: Đặc tả kỹ thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.6: Tiêu chí Hoàn thành Kỹ thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.7: Tiêu chí Nghiệm thu Sản phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.8: Nhiều Phần, Một Cổng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.9: Ví dụ

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.10: Ảnh xạ vào Quy trình của Bạn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

12.11: Tóm tắt

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 13: Định nghĩa Hoàn thành Tùy chỉnh (RMF 2)

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.1: Nguyên tắc: Mỗi Hạng mục Công việc là Duy nhất

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.2: Hành vi: Duy trì Một hoặc Nhiều Mẫu DoD

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.3: Hoạt động: Xác định Mẫu Định nghĩa Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.4: Duy trì và Cải thiện Mẫu Định nghĩa Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.5: Nhiều Mẫu Định nghĩa Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.6: Hành vi: Sử dụng Mẫu làm Điểm Khởi đầu cho Định nghĩa Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.7: Hành vi: Đồng ý về Định nghĩa Hoàn thành Tùy chỉnh

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.8: Hoạt động: Định nghĩa Định nghĩa Hoàn thành cho Hạng mục Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.9: Một Phần Mở rộng Khác cho Quy trình làm việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.10: Hành vi: Hoàn thiện DoD trước khi Bắt đầu Thực hiện

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.11: Hoạt động: Phân tích Ngoại tuyến để Hoàn thiện DoD của PBI

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.12: Thêm việc Hoàn thiện vào Quy trình

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.13: Hành vi: Theo dõi Mức độ Hoàn thành trong Hạng mục Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.14: Thêm Theo dõi Tiến độ

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.15: Hành vi: Kiểm soát Công việc theo Mức độ Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.16: Hoạt động: Sử dụng DoD để Xác định Mức độ Hoàn thành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.17: Cách Thức Vận Hành của Kiểm Soát

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

13.18: Tóm lại

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 14: Cuộc sống Với RMF 1 & 2

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.1: Chi phí

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.2: Khung thời gian

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.3: Tác động đến Nhóm thực thi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.4: Tác động đến Chủ Sản Phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.5: Tác động đến Ban Lãnh đạo

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

14.6: Tổng kết

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 15: Cài đặt RMF 2

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.1: Sự Tham gia của Các Bên liên quan

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.2: Mức độ Chi tiết

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.3: Thỏa thuận Làm việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.4: Công việc Ban đầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.5: Triển khai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

15.6: Tóm tắt

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phần IV: Triển khai có kiểm soát

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 16: Yêu cầu Cuối cùng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.1: Nó Đã Ở Đó Suốt Thời Gian Qua

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.2: Sức mạnh của Thời điểm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.3: Đảo Ngược Cách Nhìn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.4: Rủi ro & Chi phí

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.5: Giá trị của việc Chờ đợi đến khi Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.6: Một Lợi ích Bổ sung

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.7: Phát biểu Vấn đề

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.8: Nhu cầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

16.9: Tóm tắt

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 17: Bối cảnh về Định nghĩa về Sự sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.1: Sự cho phép làm việc trong Lean

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.2: Tiêu chí đầu vào của cột trong Kanban

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.3: Scrum và các Quy trình Agile khác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.4: Lướt sóng > Viết code

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.5: Giải pháp Điểm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

17.6: Chúng ta Đã Sẵn sàng để Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 18: Xác định Định nghĩa về Sự sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.1: Mục đích

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.2: Tùy chỉnh

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.3: Cấu trúc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.4: Ví dụ

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.5: Sự đồng thuận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.6: Kiểm soát

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

18.7: Tóm tắt

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 19: Định nghĩa Sẵn sàng Tùy chỉnh (RMF 3)

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.1: Thêm một Cổng trong Quy trình

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.2: Cấu trúc của một Định nghĩa Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.3: Tiêu chí Xuất phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.4: Tiêu chí Tiếp nhận Kỹ thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.5: Không Có Hại Gì Khi Trùng Lặp

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.6: Hành vi: Duy trì Một hoặc Nhiều Mẫu DoR

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.7: Tại sao Cần có Mẫu Định nghĩa về Sự sẵn sàng?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.8: Hoạt động: Xác định Mẫu DoR

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.9: Duy trì Mẫu DoR theo thời gian

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.10: Mẫu chỉ là Điểm khởi đầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.11: Hành vi: Thống nhất Định nghĩa Sẵn sàng Tùy chỉnh

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.12: Hoạt động: Xác định Định nghĩa Sẵn sàng cho Hạng mục Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.13: Hành vi: Đảm bảo các Hạng mục Sẵn sàng trước khi Bắt đầu Thực hiện

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.14: Các Điều kiện Ngoài Tầm Kiểm soát của Nhóm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.15: Hành vi: Theo dõi Mức độ Sẵn sàng trong Hạng mục Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.16: Hành vi: Kiểm soát Công việc theo Mức độ Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.17: Hoạt động: Sử dụng DoR để Xác định Mức độ Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

19.18: Tóm lại

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phần V: Tổng hợp

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 20: Hầu hết Thời hạn Đầu Không Quan trọng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.1: Thời hạn Thực sự

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.2: Thời hạn Tùy tiện Không Quan trọng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.3: Hãng Hàng không

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.4: Nguồn Gốc của Nợ Kỹ Thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.5: Thời Hạn Tiếp Thị và Bán Hàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.6: Thời hạn dự án

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.7: Có một cách khác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.8: Thời hạn võ đoán là không cần thiết

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.9: Thời hạn võ đoán phải được loại bỏ

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.10: Thời Hạn Thực Tế Vẫn Là Một Yếu Tố

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.11: *Pis Aller*¹

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

20.12: Kết luận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

¹Một biện pháp cuối cùng, xuất phát từ tiếng Pháp.

Chương 21: Năng lực 1: Quy trình Hoàn thiện Yêu cầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.1: Dưới Thế nào, Trên Thế ấy

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.2: Nguyên tắc: Tính minh bạch trong Mọi Công việc Cần thiết

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.3: Hành vi: Trách nhiệm Đi cùng Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.4: Hành vi: Theo dõi Trạng thái của Yêu cầu một cách Rõ ràng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.5: Hành vi: Hiện rõ Tất cả Công việc Liên quan đến Chuẩn bị và Triển khai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.6: Hoạt động: Công việc Chuẩn bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.7: Hành vi: Hiện rõ Tất cả Công việc Cần thiết để Hoàn thành một Hạng mục Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.8: Hành vi: Ưu tiên Sự sẵn sàng hơn Thời hạn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

21.9: Kết luận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 22: Cách Công việc và Thông tin Vận hành trong Scrum với RMF

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.1: Các Lưu ý Mở đầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.2: Thu thập và Chuẩn bị Yêu cầu Ban đầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.3: Khởi động Công việc Chuẩn bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.4: Lập kế hoạch và Thực hiện Công việc Chuẩn bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.5: Xem xét Kết quả Công việc Chuẩn bị

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

22.6: Lập kế hoạch và Hoàn thành Triển khai

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 23: Tác động của RMF

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

23.1: Vòng đời của một Yêu cầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

23.2: Ví dụ về Luồng Thông tin và Công việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

23.3: Trước đây

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

23.4: Sau đó

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

23.5: Lợi ích

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 24: Chuyển đổi sang RMF

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.1: Mô hình Áp dụng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.2: Tích hợp vào Quy trình Scrum

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.3: Các Framework Khác

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.4: Phản đối Chính: Hạng mục Công việc Sẵn sàng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.5: Sự Thay đổi Lớn: Tư duy

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.6: Lời khuyên cho Sự Thay đổi

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

24.7: Kết luận

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Chương 25: Phụ thuộc vào Bạn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

25.1: Tổng kết

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

25.2: Giờ là Lướt của Bạn

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phần VI: Tài nguyên

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phụ lục A: Scrum Không Phải Là Vấn Đề

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Scrum là gì?

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các khung làm việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Scrum Giải Quyết Việc Quản Lý Dự Án và Công Việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Sai lầm khi Coi Scrum như một Khung Quản lý Sản phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Không Có Cơ chế Được Quy định để Hoàn thiện Yêu cầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Không Đầu tư Chuyên môn Kỹ thuật vào Việc Phát triển Yêu cầu

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các Mẫu hình Phản tác dụng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Cần có Sự Mở rộng

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Bổ sung cho Scrum

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phụ lục B: Khuôn khổ Synapse™

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Những gì Synapse Bao gồm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Ba Sự Thông thạo

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Cách Áp dụng Synapse

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Kết hợp hai khung làm việc

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Cấu trúc của Khung làm việc Synapse

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các kỹ năng chuyên sâu của tổ chức

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các năng lực của tổ chức

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các thói quen của tổ chức

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Cấu trúc của Khung làm việc Synapse

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Cấu tạo của một Thực hành

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Tầm quan trọng của trật tự

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Tác động của Synapse đối với Cuốn sách này

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phụ lục C: Những Phản Đối và Trở Ngại Phổ Biến đối với RMF 1

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Những Phản Đối Phổ Biến

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Các Trở ngại Phổ biến

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phụ lục D: Danh sách Tiêu chí Khởi đầu của DoD

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Danh sách Khởi đầu cho Tiêu chí Đạt chuẩn Kỹ thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Danh sách khởi đầu về Tiêu chí đầu vào sản phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Phụ lục E: Danh sách Tiêu chí Khởi đầu cho DoR

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Tiêu chí Hoàn thành Sản phẩm

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Tiêu chí đầu vào kỹ thuật

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Tiêu chí Khởi động Sprint

Nội dung này không có trong phiên bản mẫu của sách. Bạn có thể mua sách trên Leanpub tại <https://leanpub.com/ready-vi>.

Mục lục

- Agile, 24
- AI, 20
- API, 22
- ban lãnh đạo, 9
- Barreras, Tom, vii
- biến chiến thắng thành thất bại, 18
- Bước Ngoặt Lớn, 12
- Central Oregon, vii
- Chi Phí Tích Lũy, 24
- chu kỳ lặp, 21
- chuyên gia tư vấn, 8
- chuyên gia về lĩnh vực, 6
- chuyên gia về nghiệp vụ, 11
- chuyển đổi Agile, iv
- Chuyển đổi sang Quy trình Hoàn thiện Yêu cầu, 72
- chất lượng mã nguồn, 25
- chậm tiến độ, 24
- Chủ Sản Phẩm, 5
- code hiện có, 11
- cúoc nổi dậy kiểu Terminator, 20
- các bên liên quan, 17
- các dịch vụ phụ thuộc, 9
- Các Framework Khác, 72
- các hạng mục tồn đọng, 24
- các nhóm kỹ thuật, 8
- các đội kỹ thuật, 9
- Công việc Ban đầu, 51
- công việc phụ thêm, 19
- Công việc Sẵn sàng Rõ ràng, 33
- cơ sở hạ tầng tài chính quốc gia Hoa Kỳ, 2
- cơ sở kiến thức, 8
- Cấu trúc của Định nghĩa Sẵn sàng, 60
- cổng thanh toán khoản vay, 3
- Cổng trong Quy trình, 60
- Definition of Done, 22
- dịch vụ upstream, 12
- ghép hình, 21
- Giám đốc kỹ thuật, 4
- giả định, 24
- giải pháp C#/.NET, 3
- giải pháp low-code, 3
- giữ chân nhân tài, 25
- Hoàn thiện DoD, 47
- hoàn thành hạng mục công việc, 18
- Hành vi: Hiện rõ Tất cả Công việc Liên quan đến Chuẩn bị và Triển khai, 68
- Hành vi: Theo dõi Trạng thái của Yêu cầu một cách Rõ ràng, 68
- hạng mục công việc, 24
- Hạng mục công việc chuẩn bị, 33
- Hạng mục Product Backlog, 12, 14
- Hạng mục Tồn đọng Sản phẩm, 7
- Hầu hết Thời hạn Deadline Không Quan trọng, 65
- hệ thống, 9
- hệ thống cũ, 6
- học thuyết lập trình giữa thế kỷ 20, 7
- Kanban, i
- kỹ năng nhóm, i
- kỹ thuật phát triển phần mềm, 7
- kỹ thuật viết yêu cầu, 9
- luồng công việc, 13
- làm lại, 17
- Làm thế nào bạn biết khi nào thì xong?, 22
- lãng phí, 24
- lòng tin, 24

- linh vực, 9
- Lỗi hỏng bảo mật, 20
- lỗi, 24
- Mô hình thác nước, 16
- Mô hình Áp dụng, 72
- môi trường phát triển, 9
- mạ vàng, 20
- Mất mát hoặc hỏng dữ liệu, 19
- mẫu thiết kế, iv
- Mức độ Chi tiết, 51
- ngân hàng, 10
- ngôn ngữ tự nhiên, 16
- Người kiểm thử, 12
- người thực hiện, 4
- nhà phát triển phần mềm, 19
- Nhân sự có thể thay thế cho nhau, 23
- nhồi nhét, 8
- nền tảng mã nguồn, 25
- nợ kỹ thuật, 65
- OutSystems, 3
- Phát triển Dựa trên PKB, iv
- Phát triển Hướng Hành vi, 7
- Phát triển hướng hành vi, iv
- phát triển phần mềm, 14, 18, 21
- phát triển theo lô lớn, 16
- Phản đối Chính, 72
- Producure, iii
- Product Backlog Item, 10, 21
- Product Owner, 9, 19
- Product và Engineering, 14
- Project Management Institute, 2
- Quy trình Hoàn thiện Yêu cầu, i, iii, iv, 67
- quy trình kỹ thuật, 9
- quy trình làm việc, 34, 46
- Quản Lý Sản Phẩm, 5
- Ready, i, ii
- Requirements Maturation Flow, 26, 69
- RMF 1, 30, 80
- RMF 2, 51
- rủi ro, 18
- Scrum, 5, 16, 22, 69
- Sprint, 10, 15, 19, 22, 24
- Sản phẩm, vai trò của, 4
- sẵn sàng*, 12
- sự hiểu biết chung, 34
- sự hiểu biết chung đã được xác nhận, 16
- Sự Tham gia của Các Bên liên quan, 51
- test-development, vii
- thay đổi giữa Sprint, 6
- The Bank, 12
- Theo dõi Mức độ Hoàn thành, 47
- thị phần, 20
- Thỏa thuận Làm việc, 51
- thời gian lãng phí, 20
- Thời gian ngừng hoạt động, 20
- thời hạn, 22
- Thời hạn dự án, 66
- thực hành kỹ thuật, 15
- Tiêu chí Tiếp nhận Kỹ thuật, 60
- Tiêu chí Xuất phẩm, 60
- Tiếp thị và Bán hàng, 65
- Triển khai, 51
- Triển khai có kiểm soát, 53
- trách nhiệm giải trình và khả năng truy vết, 14
- Tuyên Bố Thành Công, 37
- tài chính, 10
- Tài nguyên, 75
- tài nguyên máy chủ, 23
- Tích hợp vào Quy trình Scrum, 72
- Tạo Điều Kiện cho Sự Sẵn Sàng, 29
- Tổ chức Sản phẩm, 4
- Tổ chức Tín dụng Liên bang, 2
- từ tự đối lập, 15
- Vi phạm quy định, 20
- viết yêu cầu phần mềm, 15
- vòng phản hồi, 13
- văn hóa, 9

yêu cầu sai, 17

yêu cầu thực sự, 19

Điều Gì Đó Còn Thiếu, 1

Đầu vào rác, đầu ra rác, 15

Định nghĩa Hoàn thành, 46

Định nghĩa Hoàn thành của Hạng mục Tồn

động Sản phẩm, 47

Định nghĩa Hoàn thành Tùy chỉnh, 45

Định nghĩa Sẵn sàng Tùy chỉnh, iii, 60

Đội Phát triển, 19

đếm số mảnh ghép, 21

định nghĩa 'hoàn thành', 19

đội ngũ lãnh đạo, 6