



Tư duy **Agile** dành cho lãnh đạo

Mục lục

Phần 1: Tại sao lãnh đạo doanh nghiệp cần có Tư duy Agile?	3
Phần 2: Mục tiêu khi đọc tài liệu này	4
Phần 3: Nội dung chính	5
3.1. Đặc điểm của nền kinh tế hiện nay	5
3.2. Khả năng linh hoạt	6
3.3. Tư duy Agile	8
3.4. Khi nào sử dụng Tư duy Agile, khi nào không?	9
3.5. Cấu trúc của một công ty Agile	14
3.6. Vai trò của lãnh đạo trong tổ chức Agile	16
3.7. Nhóm tự quản liên chức năng	16
Tài liệu tham khảo	20

Phần 1: Tại sao lãnh đạo doanh nghiệp cần có Tư duy Agile?

Môi trường kinh doanh biến động ngày nay hầu hết các lãnh đạo doanh nghiệp đều ưu tiên gia tăng khả năng linh hoạt [1].

Để tổ chức đạt được sự linh hoạt này, không chỉ khối chiến lược hay nghiên cứu phát triển thực hiện nhiệm vụ đổi mới mà đó còn yêu cầu từng bộ phận và nhóm linh hoạt, sáng tạo để thích ứng và đổi mới dịch vụ, sản phẩm phù hợp với nhu cầu của khách hàng bên ngoài và nội bộ. Phần lớn ngân hàng lớn trên thế giới đã lựa chọn tư duy Agile và phương pháp làm việc Agile để vừa duy trì sự hiệu quả, an toàn mà còn là sự nhanh nhạy không chỉ cho khối công nghệ thông tin mà cả kinh doanh và vận hành [2].

Để chuyển đổi số thành công mang lại sự hiệu quả hơn của vận hành hay những mô hình kinh doanh số mới, tổ chức cần áp dụng tư duy Agile ở mức độ tổ chức [3]. Và dĩ nhiên khi chúng ta nói ở mức độ tổ chức là chúng ta nói tới sự dẫn dắt của khối kinh doanh. Điều đó dẫn tới lãnh đạo của khối kinh doanh vận hành có Tư duy Agile là điều kiện để trở thành một tổ chức số.

Phần 2: Mục tiêu khi đọc tài liệu này

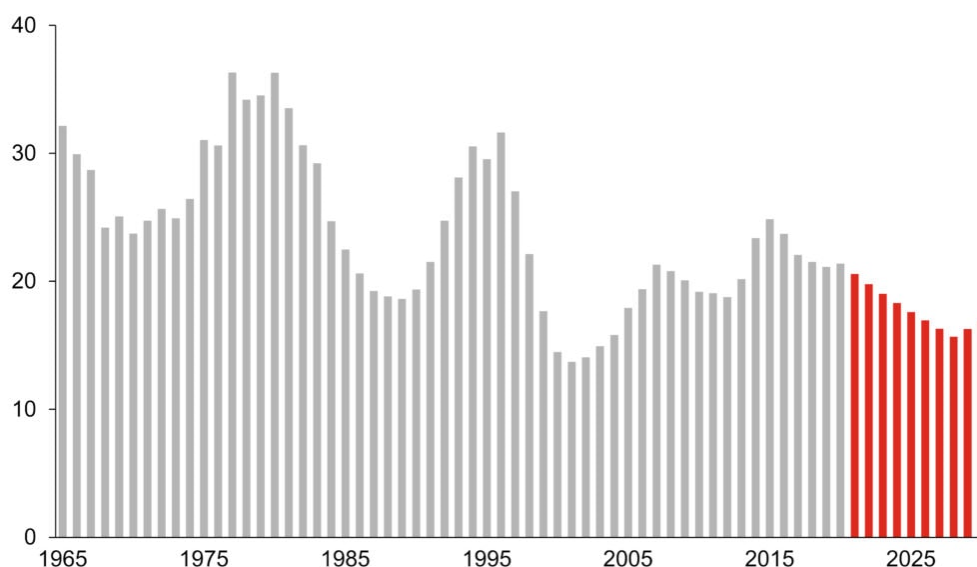
- Nhận thức được đặc điểm biến động của việc kinh doanh trong thời đại ngày nay, cuộc cách mạng số.
- Liệt kê được các mức linh hoạt từ hoạt động vận hành của nhóm tới chiến lược đổi mới để giúp doanh nghiệp duy trì và phát triển.
- Hiểu được Tư duy Agile là một cách thức để đạt được năng lực linh hoạt.
- Nhận biết được khi nào sử dụng Tư duy Agile, khi nào không?
- Hiểu được các phương pháp phổ biến của Tư duy Agile trong phát triển dịch vụ, sản phẩm, dự án và quản lý đội ngũ.
- Hiểu được cách cấu trúc một tổ chức Agile để duy trì sự an toàn, ổn định cũng như đảm bảo sự linh hoạt.
- Hiểu được sự chuyển dịch vai trò, trách nhiệm của lãnh đạo trong một tổ chức Agile khác với một tổ chức truyền thống.

Phần 3: Nội dung chính

3.1. Đặc điểm của nền kinh tế hiện nay

Môi trường kinh doanh ngày một khắc nghiệt, thời gian trung bình một doanh nghiệp ở trong S&P 500 giảm từ 30-35 năm trong thập niên 70 giờ xuống còn 15-20 năm (Chart 1)

Chart 1: Average company lifespan on S&P 500 Index in years (rolling 7-year average)



Sự khắc nghiệt của môi trường kinh doanh được mô tả trong cụm từ VUCA

Volatility (biến động) - Biến động đề cập đến tốc độ thay đổi trong một ngành, thị trường hoặc thế giới nói chung. Nó gắn liền với những biến động về nhu cầu, sự hỗn loạn và thời gian ngắn của thị trường và nó đã được ghi lại đầy đủ trong các tài liệu về tính năng động của ngành. Thế giới càng biến động, mọi thứ càng thay đổi nhanh chóng.

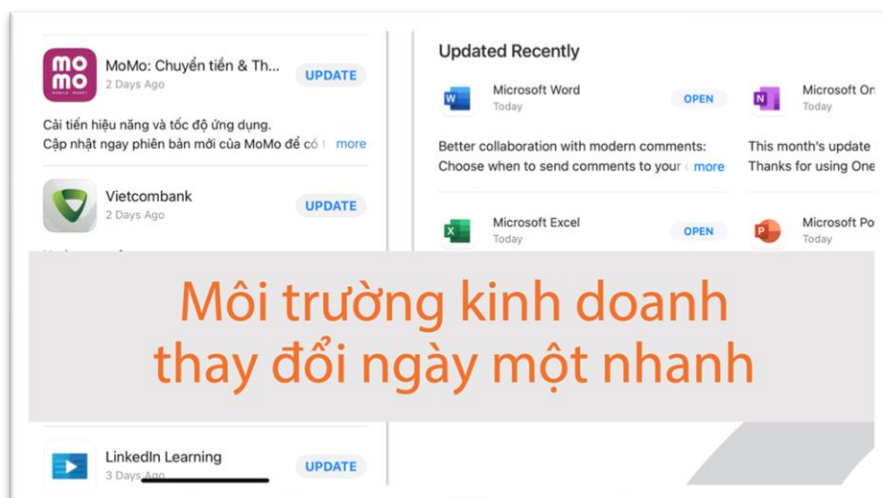
Uncertainty (không chắc chắn) - Sự không chắc chắn đề cập đến mức độ mà chúng ta có thể tự tin dự đoán tương lai. Một phần của sự không chắc chắn được nhận thức và liên quan đến việc mọi người không thể hiểu được điều gì đang xảy ra. Tuy nhiên, sự không chắc chắn cũng là một đặc tính khách quan hơn của môi trường. Môi trường thực sự không chắc chắn là những môi

trường không cho phép bất kỳ dự đoán nào, cũng không dựa trên cơ sở thống kê. Thế giới càng bất định, càng khó dự đoán.

Complexity (Phức hợp) - Độ phức tạp đề cập đến số lượng các yếu tố mà chúng ta cần tính đến, sự đa dạng của chúng và mối quan hệ giữa chúng. Càng nhiều yếu tố, sự đa dạng của chúng càng lớn và chúng càng liên kết với nhau, môi trường càng phức tạp. Dưới độ phức tạp cao, không thể phân tích đầy đủ môi trường và đưa ra kết luận hợp lý. Thế giới càng phức tạp thì càng khó phân tích.

Ambiguity (Mơ hồ) - Sự mơ hồ đề cập đến sự thiếu rõ ràng về cách giải thích điều gì đó. Một tình huống không rõ ràng, ví dụ, khi thông tin không đầy đủ, mâu thuẫn hoặc quá không chính xác để đưa ra kết luận rõ ràng. Nói một cách tổng quát hơn, nó đề cập đến sự mờ nhạt và mơ hồ trong các ý tưởng và thuật ngữ. Thế giới càng mơ hồ, càng khó giải thích.

Nếu chúng ta tham khảo các ứng dụng, dịch vụ mà chúng ta dùng hằng ngày như trên điện thoại của mình hay các dịch vụ được cập nhật trên web thì chúng ta có thể thấy chúng được cập nhật hằng ngày hoặc hằng tuần (Hình dưới)



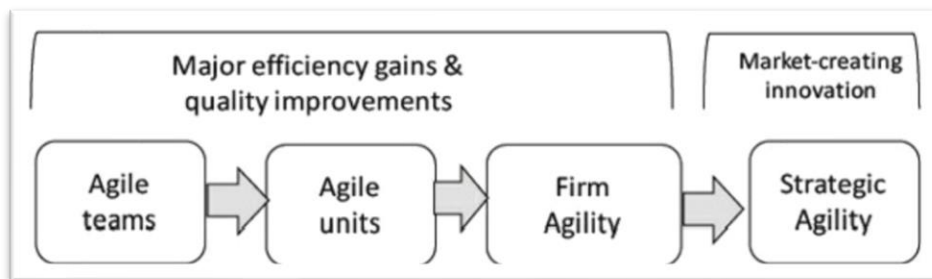
3.2. Khả năng linh hoạt

Vậy chúng ta cần năng lực gì để thích ứng với môi trường kinh doanh này? Tổ chức Agile Business Consortium đã định nghĩa năng lực linh hoạt:

Thích ứng nhanh chóng với sự thay đổi của thị trường – bên trong & bên ngoài

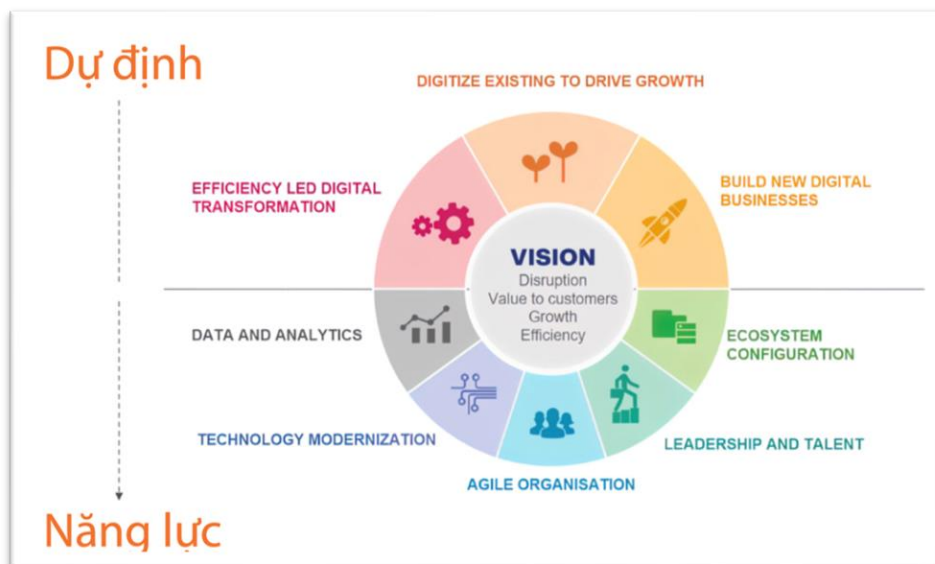
- Phản hồi nhanh chóng và linh hoạt với nhu cầu của khách hàng
- Thích ứng và dẫn dắt sự thay đổi theo cách vẫn đạt được năng suất, hiệu quả về chi phí mà không hy sinh chất lượng.
- Liên tục duy trì được lợi thế cạnh tranh.

Khi xét năng lực ở những quy mô khác nhau chúng ta sẽ nhìn thấy những hiệu quả khác nhau. Steve Denning trong tác phẩm The Age of Agile đã chia năng lực ra những cấp độ khác nhau:



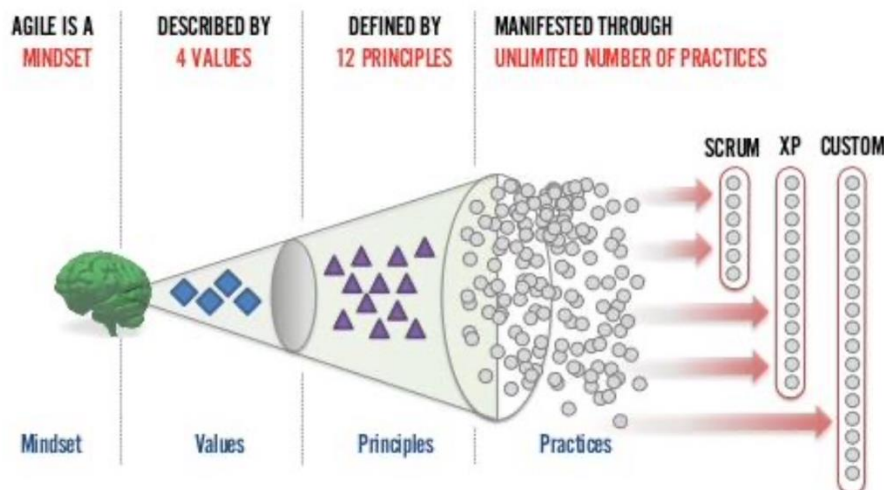
- Mức vận hành: để cải thiện hiệu suất, chi phí và gia tăng chất lượng sản phẩm, dịch vụ đáp ứng nhu cầu khách hàng ở mức độ nhóm, đơn vị hay toàn bộ tổ chức cho những dịch vụ và sản phẩm hiện thời.
- Mức độ chiến lược: Những đổi mới để tạo ra những mô hình kinh doanh mới.

Và đặc biệt để chuyển đổi số thành công mang lại sự hiệu quả hơn của vận hành hay những mô hình kinh doanh số mới, tổ chức cần áp dụng tư duy Agile ở mức độ tổ chức [3] như hình dưới.



3.3. Tư duy Agile

Phần lớn các tổ chức, mà cụ thể là các ngân hàng đã lựa chọn Agile như cách thức để đạt được khả năng linh hoạt. Ahmed Sidky - Chủ tịch của ICAgile có một cách định nghĩa Agile như ở hình dưới.



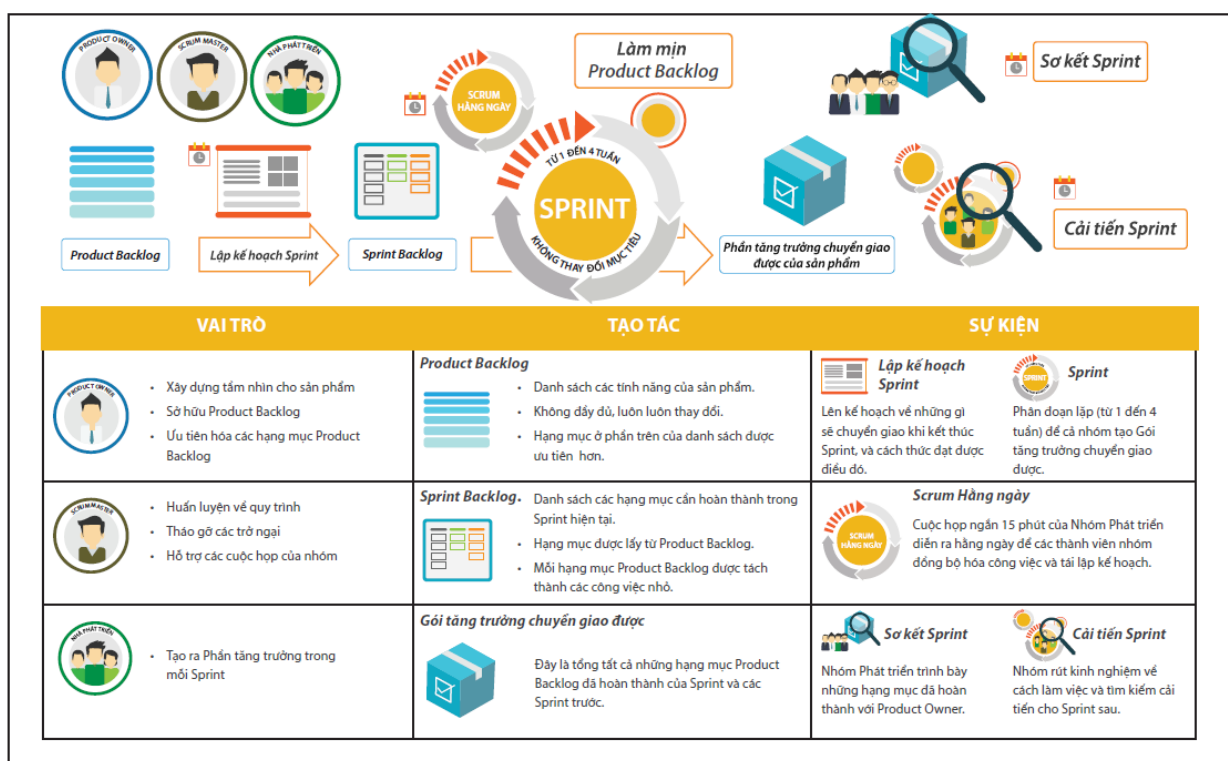
Như vậy chúng ta có thể nói Agile là cách tư duy được định nghĩa thông qua bốn điểm được đánh giá cao:

- 1) Trách nhiệm và tự chủ của mỗi cá nhân và đội nhóm hơn là kiểm soát, tuân thủ quy trình, công cụ.
- 2) Sản phẩm dịch vụ tới khách hàng hơn là những bộ tài liệu dễ hiểu.
- 3) Cộng tác với khách hàng để cùng kiến tạo giá trị.
- 4) Phản hồi với thay đổi để tạo kết quả hơn là chỉ tuân theo kế hoạch.
- 5) Có rất nhiều kỹ thuật khác nhau đã được tạo ra thể hiện tinh thần đó như:
- 6) Chia thành các nhóm nhỏ tự quản với một nhiệm vụ cụ thể gọi là nhóm đặc nhiệm – squad.
- 7) Cả nhóm hằng ngày cùng họp đứng để chia sẻ thông tin và điều chỉnh hành động để bám sát mục tiêu chứ không đợi tới họp giao ban hay chỉ báo cáo qua email.
- 8) Trong một khoảng thời gian ngắn như hai hay vài tuần, cả đội không phân biệt kỹ năng và chức danh cùng làm việc để hoàn thành sản phẩm cho khách hàng chứ không chỉ làm rõ yêu cầu khách hàng.

- 9) Thử nghiệm với khách hàng thường xuyên để xác định xem tiếp theo nên đi theo hướng nào dù kế hoạch là rất dài.

Scrum đã tổ chức lại khá nhiều kỹ thuật nhỏ đó để đưa ra một cách thức làm việc cho phần lớn các nhóm Agile. Tuy nhiên Scrum chỉ là một cách thức tổ chức, còn rất nhiều cách thức khác nữa.

Hình dưới tóm tắt phương pháp Scrum trích từ Cẩm nang Scrum:



3.4. Khi nào sử dụng Tư duy Agile, khi nào không?

Câu hỏi mà rất nhiều người đặt ra là liệu Agile có phù hợp với đặc thù công việc của đơn vị mình?

Những người có kinh nghiệm thấy ngay rằng không thể trả lời nhanh câu hỏi này. Nhưng chúng ta luôn có những manh mối để suy nghĩ và bắt đầu.

Hãy xuất phát từ phạm vi hẹp: Quản trị dự án. Các giáo sư dạy môn quản trị dự án hiện đại nêu đại khái: Có hai loại dự án cần những cách tiếp cận khác nhau, một là các dự án thuộc loại Thực thi (Execution), một loại ở thái cực kia là các dự án thuộc loại Mới tinh (Novel). Loại Thực thi thì

cần lên kế hoạch kĩ, chi tiết, rồi thi hành và kiểm soát thật chặt để đạt được kết quả tốt. Loại Mới tinh thì lại không thể làm thế vì nó ẩn chứa quá nhiều yếu tố không đoán được (uncertainty). Với loại này, một kế hoạch cần có khả năng cập nhật với các phản hồi mới, dữ liệu mới trong quá trình triển khai dự án. Do đó nó nên được thực hiện với khả năng thích ứng tốt, có vòng phản hồi ngắn, và cần sự cộng tác chặt chẽ giữa các bên liên quan, đội dự án cũng được linh hoạt hơn, tự quản nhiều hơn thay vì làm việc thuần túy kiểu “ra lệnh và kiểm soát” (command-and-control). Nói gọn lại, các dự án thuộc típ Thực thi sẽ vẫn hiệu quả tuyệt vời khi chúng ta triển khai theo dạng truyền thống; còn các dự án loại “mới tinh” thì nên sử dụng Agile mới ổn. Chuyện có vẻ rõ như ban ngày.

Tuy nhiên, nhận ra dự án thuộc loại Thực thi hay Mới tinh không bao giờ là hiển nhiên. Một dự án được con người vận hành, luôn tiềm ẩn những yếu tố khó đoán định trong quá trình tương tác nội bộ, tương tác trong-ngoài, tương tác với môi trường; kết quả thì không phải lúc nào cũng dễ dàng hình dung rõ ràng ngay từ lúc lập kế hoạch ban đầu để có thể lên kế hoạch chi tiết và xây dựng cho được một biểu đồ Gantt đẹp đẽ. Còn khi thực thi, môi trường có thể vẫn cứ biến động không ngừng. Hóa ra, nhìn nhận đâu là dự án kiểu gì không hề dễ dàng.

Nói vậy không có nghĩa là giới học thuật và thực hành chưa có những nỗ lực đáng kể để giúp tình hình bớt bế tắc. Trong bài báo mới đây (Embrace Agile) đăng trên HBR, đồng tác giả Scrum, ông Jeff Sutherland cùng với giáo sư Takeuchi ở Đại học Harvard và Darrell K. Rigby đã cung cấp một số lời khuyên hữu ích. Các ông cho rằng Agile phù hợp khi:

- Nhu cầu khách hàng và yêu cầu về giải pháp thay đổi thường xuyên.
- Cần cộng tác chặt chẽ với khách hàng và có thể cung cấp các phản hồi nhanh, khách hàng nắm rõ hơn về những gì họ mong muốn.
- Vấn đề rất phức tạp, giải pháp không rõ từ đầu và phạm vi cũng không được xác định rõ.
- Đặc tả sản phẩm có thể thay đổi, và những sáng tạo đột phá luôn được ưu tiên.
- Cộng tác liên-chức năng là sống còn.
- Việc phát triển tăng trưởng mang lại giá trị, và khách hàng có thể sử dụng ngay những giá trị này.
- Công việc có thể bẻ nhỏ thành từng phần và có thể được thực thi trong những phân đoạn lặp ngắn.
- Những thay đổi ở phút chót có thể quản lí được.
- Những sai sót có thể mang lại những bài học chứ không mang đến những thảm họa.

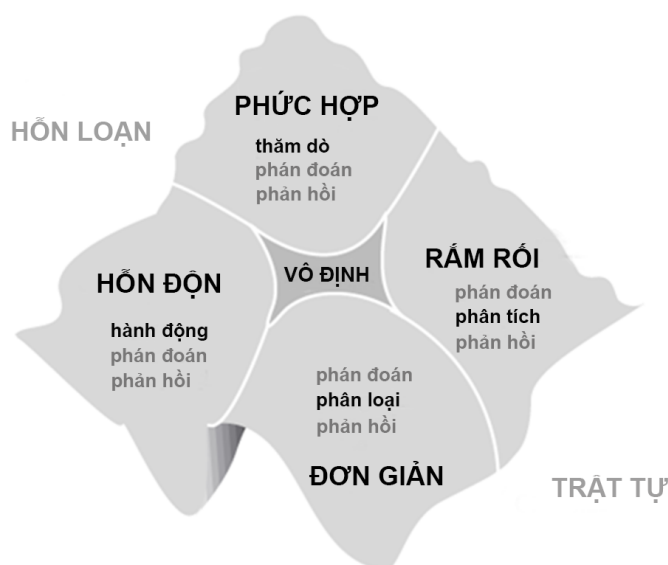
Và, Agile có thể không phù hợp trong những điều kiện ngược lại:

- Điều kiện thị trường ổn định và có thể tiên lượng.
- Yêu cầu rất rõ ràng và luôn ổn định.
- Khách hàng không thể cộng tác thường xuyên.

- Công việc tương tự những gì đã làm trước đó, và giải pháp là rất rõ ràng. Đặc tả chi tiết có thể làm ra với sự dự đoán rõ ràng và chính xác. Vấn đề có thể giải quyết tuần tự qua từng bộ phận chức năng mà không gặp trở ngại nào.
- Khách hàng không thể bắt đầu kiểm thử các phần sản phẩm cho tới khi sản phẩm hoàn chỉnh.
- Thay đổi phút chót rất tốn kém, hoặc không thể.
- Sai sót trong thực thi có thể dẫn đến thảm họa không thể cứu vãn được.

Trong khi những lời khuyên chí tình này xuất phát từ những bộ óc có thừa trải nghiệm với Agile. Trong số họ có cả cha đẻ của Scrum, và "ông nội" của Scrum (ông Takeuchi, đồng tác giả bài báo trứ danh đánh dấu sự ra đời ý tưởng Scrum năm 1986 cũng trên Harvard Business Review, bài báo "The New New Product Development Game"), thì nó cũng chỉ cung cấp lời khuyên kiểu "rule of thumb" cho những người thực hành, mà có thể chưa làm thỏa mãn những người muốn có cái nhìn mang tính kỹ thuật để ra những quyết định quan trọng. Thật may mắn là chúng ta có thể tìm thấy một chút manh mối với mô hình Cynefin, một mô hình nghiên cứu về phức hợp.

Trong bài báo "A Leader's Framework for Decision Making" trên HBR số Tháng 11 năm 2007, các tác giả Snowden và Boone trình bày một mô hình phân loại thế giới trong các vùng với các đặc điểm tương đối khác nhau với độ phức tạp tăng dần: Hiển nhiên (Obvious), Rắc rối (Complicated), Phức hợp (Complex) và Hỗn loạn (Chaotic). Nếu các dự án tính phức tạp là Hiển nhiên hoặc Rắc rối, thì chúng thuộc loại Thực thi đã đề cập ở đầu bài này. Với các dự án có độ phức tạp từ Phức hợp đến Hỗn loạn, chúng ta có loại Mới tinh. Dĩ nhiên, chúng đòi hỏi những cách thức suy nghĩ và ra quyết định khác nhau.



Cynefin framework

Bảng dưới đây tóm tắt những dấu hiệu nhận dạng độ phức tạp cho các bối cảnh khác nhau. Dưới góc nhìn của Cynefin, không có một chiến lược ra quyết định kiểu vạn năng cho các tình huống. Chúng ta cần nhiều hơn một kiểu nghĩ để sử dụng trong các tình huống khác nhau.

Tình huống	Đặc điểm nhận dạng	Vai trò người ra quyết định
Hiển nhiên	<ul style="list-style-type: none"> Các khuôn mẫu (pattern) và sự kiện lặp đi lặp lại Mọi người đều nhìn rõ mối quan hệ nhân-quả Có một câu trả lời đúng đắn Mọi người biết rõ những cái đã biết Quản lí theo thực tế(Fact-based management) 	<ul style="list-style-type: none"> Phán đoán trước, phân loại sau, rồi phản hồi Sử dụng quy trình đúng đắn Ủy quyền Sử dụng kinh nghiệm tốt (best practice) Truyền đạt rõ ràng, trực tiếp Giao tiếp hai chiều có thể không cần thiết
Rắc rối	<ul style="list-style-type: none"> Cần chuyên gia phân tích Có thể nhận biết quan hệ nhân-quả nhưng không hiển nhiên đối với mọi người Có nhiều hơn một câu trả lời Mọi người biết là có những thứ chưa biết đến Quản lí theo thực tế 	<ul style="list-style-type: none"> Phán đoán trước, phân tích sau, rồi phản hồi Thiết lập đội chuyên gia Lắng nghe các lời khuyên có phần khác biệt
Phức hợp	<ul style="list-style-type: none"> Thay đổi liên tục và không thể tiên lượng Không có câu trả lời đúng đắn; các khuôn mẫu hoạt động được ló dần ra 	<ul style="list-style-type: none"> Thử nghiệm trước, phán đoán sau, rồi phản hồi Tạo môi trường thử nghiệm cho phép các khuôn mẫu xuất hiện.

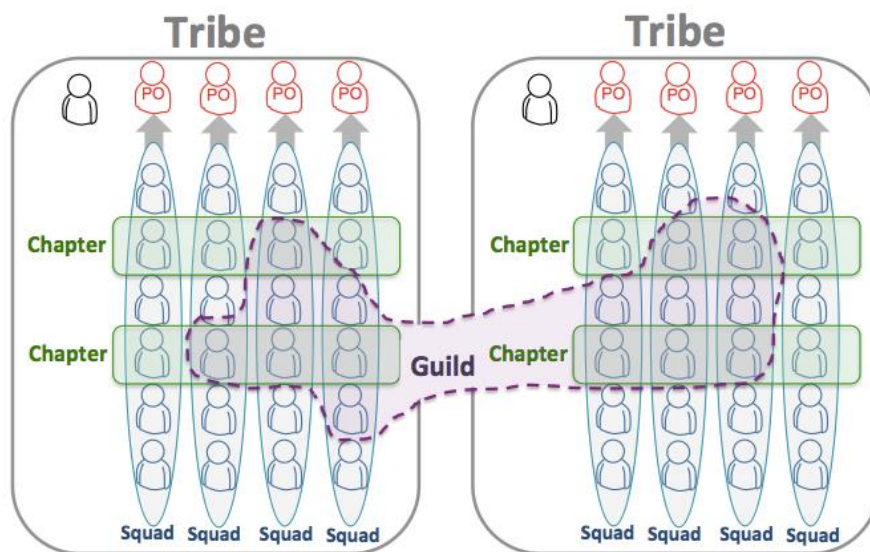
	<ul style="list-style-type: none"> • Mọi người không biết những điều chưa biết • Rất nhiều ý tưởng đối nghịch nhau • Cần những tiếp cận sáng tạo • Lãnh đạo dựa-trên-khuôn-mẫu (pattern-based leadership) 	<ul style="list-style-type: none"> • Tăng mức độ tương tác và giao tiếp hai chiều • Dùng các phương pháp để tạo lập thật nhiều ý tưởng: thảo luận mở; tạo lập biên; khuyến khích các nhân tố thu hút (attractor); khuyến khích sự bất đồng quan điểm và sự đa dạng; quản lí các điều kiện ban đầu và giám sát để hình thành các khuôn mẫu.
Hỗn độn	<ul style="list-style-type: none"> • Nhiều loạn cao độ • Không rõ quan hệ nhân- quả, thậm chí không có điểm nào để tìm kiếm manh mối • Không thể biết được điều gì • Phải ra nhiều quyết định cấp tập mà không có thời gian để suy nghĩ • Lãnh đạo dựa-trên-khuôn-mẫu (pattern-based leadership) 	<ul style="list-style-type: none"> • Hành động trước, phán đoán sau, rồi phản hồi • Quan sát xem cái nào hiệu quả thay vì tìm kiếm câu hỏi đúng đắn • Hành động tấp lự để văn phòng trật tự (mệnh lệnh và kiểm soát) • Truyền đạt rõ ràng, trực tiếp

Nhận biết và ra quyết định phù hợp với độ phức tạp, theo Snowden và Boone.

Như chúng ta thấy, Agile phù hợp hơn ở vùng Phức hợp (hai cha đẻ của Scrum, Ken Schwaber và Jeff Sutherland cũng đã nhắc lại nhiều lần), nơi có thể vận dụng cách thức thí nghiệm-phán đoán-phản hồi để ra quyết định, trao quyền tự quản cho nhóm, thiết lập các điều kiện ban đầu và luôn tìm kiếm các khuôn mẫu thích ứng có tính tối ưu nổi lên trong quá trình nhóm làm việc tự-tổ-chức. Ở một mức độ nhất định, Agile có thể phù hợp ở vùng Rối loạn, nhưng nói chung, vùng Phức hợp là vùng dễ thấy tính tương thích nhiều nhất.

3.5. Cấu trúc của một công ty Agile

Cấu trúc sử dụng ma trận khi áp dụng mô hình Spotify



3.5.1. Squad

Squad giống như một Nhóm Scrum, và được thiết kế như là một startup-nhỏ. Những người có đủ những kỹ năng và công cụ cần thiết để thiết kế, phát triển, kiểm thử và phát hành sản phẩm, sẽ ngồi lại với nhau. Họ là nhóm tự tổ chức và quyết định cách làm việc riêng - có thể dùng các Sprint trong Scrum, Kanban hoặc sự kết hợp của cả hai phương pháp.

Mỗi Squad có một nhiệm vụ dài hạn kiểu như xây dựng và cải tiến ứng dụng trên Android, tạo ra trải nghiệm radio trên Spotify, mở rộng hệ thống backend, hoặc cung cấp các giải pháp thanh toán. Hình ảnh dưới đây minh họa những trách nhiệm mà Squad phải đảm nhận ở các phân đoạn khác nhau của quá trình trải nghiệm người dùng.

Tribe là một tập hợp các squad làm việc trong những lĩnh vực có liên quan - như là Khách hàng cá nhân hay Trái phiếu hoặc vận hành.

3.5.2. Các Chapter và Guild

Đâu cũng có sự hiện diện của sự bất lợi. Sự bất lợi tiềm tàng khi muốn đạt đến sự tự quản tuyệt đối là sự suy giảm về kinh tế quy mô. Kiểm thử ở Squad A có thể đang phải vật lộn với vấn đề mà kiểm thử ở Squad B đã giải quyết tuần trước. Nếu tất cả các kiểm thử viên của các Squad và các

Tribe có thể tập hợp lại thì họ sẽ có thể chia sẻ kiến thức và tạo ra những công cụ giúp ích cho tất cả các Squad.

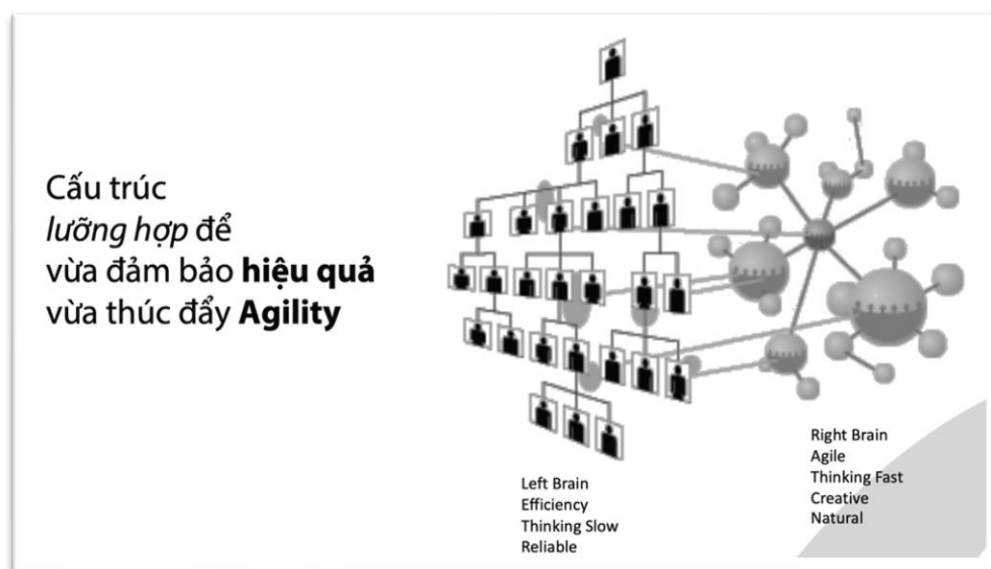
Nếu mỗi Squad có tính hoàn toàn tự chủ mà lại thiếu đi sự giao tiếp giữa các Squad, thì ý nghĩa của công ty là gì? Spotify có thể được phân chia thành 30 công ty khác nhau.

Vì vậy chúng tôi có các Chapter và Guild. Đây như là keo dính gắn kết toàn công ty, nó mang đến kinh tế quy mô mà không phải hi sinh quá nhiều sự tự chủ.

Chapter là một gia đình bé nhỏ gồm những người có kỹ năng giống nhau và năng lực tương đồng, trong một Tribe.

Guild lại là "cộng đồng quan tâm/yêu thích" có quy mô rộng và hữu cơ hơn. Đây là một nhóm của những người muốn chia sẻ kiến thức, công cụ, mã nguồn và những phương pháp thực hành. Chapter luôn nằm trong nội bộ của Tribe, trong khi Guild thường trải rộng ra toàn tổ chức.

Bên cạnh mô hình Spotify thì khi triển khai Agile trong các tổ chức truyền thống như ngân hàng thì cấu trúc lưỡng hợp của Kotter thường tỏ ra hữu ích hơn.



Trong mô hình này cấu trúc truyền thống được duy trì và mạng lưới cấu trúc các nhóm sáng tạo ít "chính thức" hơn hình thành để triển khai các sáng kiến đổi mới.

3.6. Vai trò của lãnh đạo trong tổ chức Agile

Để hiểu rõ vai trò của một lãnh đạo Agile, chúng ta xem xét cấu trúc của một nhóm Agile phổ biến nhất - Nhóm Scrum (Trích Cẩm nang Scrum - thuộc tủ sách của Học viện Agile). Nhóm Scrum bao gồm các vai trò Product Owner, Nhà Phát triển, và Scrum Master. Nhóm Scrum là nhóm tự quản (self-managed) và liên chức năng (cross-functional). Nhóm tự quản tự chọn việc sẽ làm, ai sẽ làm và cách thức tốt nhất để hoàn thành công việc chứ không bị chỉ đạo bởi ai đó bên ngoài nhóm. Nhóm liên chức năng có đủ kỹ năng cần thiết để hoàn thành công việc mà không phụ thuộc vào bất kỳ người ngoài nào khác ngoài nhóm. Mô hình nhóm trong Scrum được thiết kế để tối ưu hóa sự linh hoạt, sáng tạo và năng suất. Nhóm Scrum chuyển giao sản phẩm theo phân đoạn lặp đi lặp lại và tăng dần, tối đa hóa cơ hội cho các phản hồi. Việc chuyển giao tăng dần (incremental) các gói sản phẩm đạt tiêu chuẩn "Hoàn thành" đảm bảo một phiên bản có thể sử dụng được của sản phẩm.

Nếu phải tìm một sự so sánh, chúng ta có thể so sánh cấu trúc của nhóm Scrum với nhóm đua thuyền rồng. Product Owner là người lái, định hướng con thuyền. Các tay chèo là các nhà Phát triển. Scrum Master là người giúp giữ nhịp để nhóm đạt hiệu năng tốt nhất trong cả cuộc đua.



3.7. Nhóm tự quản liên chức năng

Nhóm Scrum có hai đặc điểm quan trọng là: tự quản (self-managed) và liên chức năng (cross-functional).

Tự quản có nghĩa là nhóm cùng ra quyết định sẽ làm gì, ai sẽ làm và làm như thế nào thông qua bàn bạc, phân công hướng đến mục tiêu chung. Trong nhóm tự quản, không ai giao việc cho ai,

mà cùng nhau chủ động lập kế hoạch, phân chia công việc hợp lý dựa trên các vai trò được định nghĩa rõ ràng từ trước. Các vai trò này được trao quyền để hoàn thành công việc hướng đến mục tiêu chung.

Liên chức năng là một cơ cấu nhóm không quá mới mẻ, và không phải lần đầu tiên xuất hiện trong Scrum. Một nhóm liên chức năng bao gồm nhiều cá nhân với các chuyên môn khác nhau đủ năng lực được kết hợp lại cùng làm việc hướng tới một mục tiêu chung. Nhóm Scrum cần được trang bị đầy đủ các kỹ năng cần thiết để hoàn thành công việc, có khả năng đạt Mục tiêu Sprint và chuyển giao phần tăng trưởng ở cuối mỗi Sprint mà không cần sự hỗ trợ từ bên ngoài.

Trong đội, các cá nhân có thể đến từ nhiều phòng ban chức năng khác nhau, cũng có thể xuất phát từ bên ngoài (khách hàng, các cá nhân có liên quan, chuyên gia tư vấn, v.v.). Nhưng khi đã thành một đội (team), thì các cá nhân làm việc tập trung cho đội như là một đơn vị (unit) để hoàn tất mục tiêu chung. Bên trong nhóm liên chức năng không có các nhóm nhỏ khác.

Ví dụ: một Nhóm Scrum "Alpha" về Content Marketing gồm 9 người được thành lập với một Product Owner, một Scrum Master, hai người có kỹ năng viết nội dung, một người có kỹ năng vẽ, một người có kỹ năng chăm sóc cộng đồng, hai người có kỹ năng làm video, một chuyên gia về nội dung, nhưng sẽ không phân chia chức năng thành các nhóm nhỏ khác như nhóm cộng đồng, nhóm video, hay nhóm content ... nữa, mà chỉ có một nhóm duy nhất "Alpha" với các cá nhân có chuyên môn khác nhau, hợp thành một đội thống nhất để làm việc hướng tới sản phẩm cần phát triển phục vụ các chiến lược marketing của công ty.

Trong cách nói của các tác giả Scrum, thì dù bạn là tester (chuyên kiểm thử) hay copywriter (chuyên viết bài) ... đều là developer (nhà phát triển), họ là nhà phát triển nhưng có chuyên môn đặc thù là kiểm thử hay viết; developer trong trường hợp này không có ý nghĩa là coder/programmer (lập trình viên). Việc xóa nhòa các chức danh công việc (Job title) này có mục đích là để hướng mọi người vào một mục tiêu chung, không phân biệt "nhãn mác" (title): phát triển sản phẩm.

Khác với nhóm liên chức năng, nhóm chức năng (functional team) thường chỉ phụ trách một loại công việc đặc thù. Ví dụ, phòng thiết kế đồ họa thì không tham gia viết nội dung, phòng nội dung thì không có ai thiết kế. Công việc của nhóm chức năng thường có tính cô lập cao. Ở những hệ thống có các phòng ban chuyên môn, công việc phát triển đòi hỏi sự cộng tác, phối hợp giữa những bộ phận khá tách biệt. Điều này làm giảm đáng kể sự linh hoạt, gây lãng phí và làm chậm tốc độ phát triển. Đối với một nhóm liên chức năng, sự cộng tác và đồng bộ diễn ra nhanh chóng và tối ưu giúp cho năng suất và hiệu quả phát triển được nâng lên.

Scrum Master là một vai trò then chốt giúp Nhóm Scrum làm việc hiệu quả. Scrum Master không trực tiếp tham gia vào công việc làm ra sản phẩm, nhưng là chất kết dính để các bên phối hợp với

nhau tạo ra sản phẩm tốt. Scrum Master không phải là quản lí của Nhóm mà là một lãnh đạo theo phong cách phục vụ (Servant Leader).

Scrum Master có trách nhiệm đảm bảo Scrum như được định nghĩa trong Hướng dẫn Scrum. Scrum Master thực hiện nghĩa vụ này bằng cách giúp các thành viên trong Nhóm Scrum và tổ chức hiểu lí thuyết và kĩ thuật Scrum.

Scrum Master có trách nhiệm với hiệu quả của Nhóm Scrum bằng cách giúp nhóm cải thiện các kĩ thuật trong khung Scrum.

Scrum Master là một lãnh đạo đích thực, người phục vụ Nhóm Scrum và tổ chức.

Với Nhóm Scrum, Scrum Master phục vụ theo nhiều cách, bao gồm những cách sau:

- Đào tạo căn bản Scrum trong trường hợp các thành viên chưa biết về khung làm việc này.
- Huấn luyện thành viên nhóm trong môi trường tự quản và liên chức năng. Ví dụ nhắc nhở luôn cập nhật Sprint Backlog và biểu đồ Sprint Burndown, thực hiện đúng sự kiện Scrum Hằng ngày. Từ đó làm cho thông tin minh bạch, từ đó các Nhà Phát triển sẽ hỗ trợ nhau tự chịu trách nhiệm với Mục tiêu Sprint.
- Giúp nhóm tập trung vào việc tạo ra những Phần tăng trưởng giá trị cao thỏa mãn Định nghĩa Hoàn thành.
- Đảm bảo tất cả các sự kiện trong Scrum diễn ra, tích cực, năng suất và trong khung thời gian. Ví dụ như chuẩn bị hậu cần thật tốt cho các sự kiện.
- Hỗ trợ tìm kiếm và sử dụng hiệu quả các công cụ hỗ trợ phát triển, chẳng hạn công cụ để quản lí Sprint Backlog, duy trì biểu đồ Sprint Burndown, các công cụ để quản lí tài liệu, họp và giao tiếp hiệu quả.
- Bảo vệ các Nhà Phát triển trước những can thiệp bên ngoài trong quá trình triển khai một Sprint. Ví dụ, Scrum Master cần giải thích rõ cho một quản lí những tác hại mà việc này gây ra cho nhóm và sản phẩm khi muốn thay đổi Mục tiêu Sprint trong quá trình đang diễn ra Sprint và nên đợi để đưa vào Sprint tiếp theo.
- Đảm bảo các Nhà Phát triển có đầy đủ các tài nguyên cần thiết phục vụ phát triển. Ví dụ như không gian làm việc, các công cụ hỗ trợ và những tài nguyên khác khi có nhu cầu, v.v...

Scrum Master phục vụ Product Owner theo nhiều cách, bao gồm những cách sau:

- Tìm kiếm các kĩ thuật để định nghĩa Mục tiêu, quản lí hiệu quả yêu cầu;
- Giúp Nhóm Scrum hiểu giá trị của các hạng mục của yêu cầu công việc rõ ràng và ngắn gọn và cách thức xây dựng chúng;
- Giúp lập kế hoạch sản phẩm theo cách thực nghiệm trong môi trường phức hợp;
- Thúc đẩy việc cộng tác với các bên liên quan theo nhu cầu hoặc khi cần thiết.

Scrum Master phục vụ tổ chức bằng nhiều cách, cụ thể như:

- Dẫn dắt, đào tạo, huấn luyện tổ chức trong việc tiếp nhận Scrum;
- Lập kế hoạch, tư vấn các cách triển khai Scrum trong tổ chức;
- Giúp đỡ nhân viên và các bên liên quan hiểu và cho phép cách tiếp cận thực nghiệm với những việc phức hợp;
- Tháo gỡ những rào cản giữa các bên liên quan và các Nhóm Scrum.

Tài liệu tham khảo

1. 5 Steps To Greater Agility In Your Organization, Hugo Moreno, Forbes, <https://www.forbes.com/sites/forbesinsights/2017/11/30/5-steps-to-greater-agility-in-your-organization>
2. A discussion on Agile in banking: Beyond buzzwords, Deepak Mahadevan, By Santiago Comella-Dorda, Belkis Vasquez-McCall, Quentin Jadoul, Francesco Di Marcello, McKinsey & Company, <https://www.mckinsey.com/industries/financial-services/our-insights/banking-matters/a-discussion-on-agile-in-banking-beyond-buzzwords>
3. Becoming An Agile Bank, Bob Meara, International Banker, <https://internationalbanker.com/banking/becoming-an-agile-bank>