Lời nói đầu

Scrum và Agile thường được coi là đồng nghĩa và có lý do chính đáng. Scrum là hiện thân của cách tiếp cận đơn giản và thuần túy nhất để quản lý công việc dự án ở cấp độ nhóm và được sử dụng bởi hơn một nửa số người thực hiện Agile trên tất cả các ngành. Hãy xem xét các số liệu thống kê dưới đây:

· Ngày nay gần 100% các tổ chức CNTT sử dụng Agile và nhiều ngành công nghiệp khác đang nhanh chóng làm theo

· Khả năng bạn sẽ tham gia một dự án Scrum hoặc giống Scrum đang nhanh chóng đạt đến 50/50 hoặc cao hơn theo thời gian.

Chúng ta sẽ bắt đầu bằng cách tìm hiểu các quy trình, vai trò, cơ chế và triết lý quản lý dự án quan trọng đằng sau Scrum. Điều này sẽ cung cấp cơ sở để hiểu Agile ở dạng thuần túy nhất của nó: Tại sao, Ai, Làm thế nào và cuối cùng là Scrum trông như thế nào được áp dụng trong thế giới thực

Scrum

Trong quản lý dự án, scrum, đôi khi được viết là Scrum hoặc SCRUM, là một framework (bộ khung) để phát triển, cung cấp và duy trì sản phẩm trong một môi trường phức tạp, [1] với trọng tâm ban đầu là phát triển phần mềm, mặc dù nó đã được sử dụng trong các lĩnh vực khác bao gồm nghiên cứu, bán hàng, tiếp thị và các công nghệ tiên tiến. [2] Nó được thiết kế cho các nhóm gồm mười thành viên trở xuống, những người chia công việc của họ thành các mục tiêu có thể hoàn thành trong vòng lặp thời gian, được gọi là **Sprint**, không quá một tháng và phổ biến nhất là hai tuần. Nhóm nghiên cứu đánh giá tiến độ trong các cuộc họp hàng ngày được đóng khung thời gian từ 15 phút trở xuống, được gọi là **daily scrums** (một hình thức họp đứng). Vào cuối sprint, nhóm tổ chức hai cuộc họp tiếp theo: đánh giá sprint thể hiện công việc đã thực hiện cho các bên liên quan để phản hồi và xem lại Sprint cho phép nhóm phản ánh và cải thiện.

## **Key ideas**

scrum áp dụng phương pháp tiếp cận thực nghiệm dựa trên bằng chứng - chấp nhận rằng vấn đề không thể được hiểu đầy đủ hoặc xác định từ trước, và thay vào đó tập trung vào cách tối đa hóa khả năng của nhóm để cung cấp nhanh chóng, đáp ứng các yêu cầu mới và thích ứng với sự phát triển công nghệ và những thay đổi trong điều kiện thị trường.

## **Scrum team**

Là một nhóm nhỏ, bao gồm **Product owner**, **Scrum master** và **Developers**. Nhóm có thể tự quản lý, liên chức năng và tập trung vào một mục tiêu tại một thời điểm: “The product goal”.

**Product owner**

Chủ sở hữu sản phẩm, đại diện cho các bên liên quan của sản phẩm và tiếng nói của khách hàng. Khả năng truyền đạt tốt là phẩm chất quan trọng nhất của chủ sở hữu sản phẩm. Nhận ra các điều cần được ưu tiên và nắm được điểm mạnh lẫn điểm yếu của các thành viên trong nhóm và các bên liên quan là rất quan trọng để chỉ đạo phát triển sản phẩm theo đúng hướng. Vai trò của **Product owner** là thu hẹp khoảng cách giao tiếp giữa nhóm và các bên liên quan, đóng vai trò là người đại diện từ cả 2 phía. Nhóm scrum chỉ nên có một **Product owner** (mặc dù **Product owner** có thể hỗ trợ nhiều hơn một nhóm)

Các nhiệm vụ chính:

* Xác định và công bố các bản phát hành.
* Thông báo tình trạng của việc triển khai và của sản phẩm.
* Chia sẻ tiến độ trong các cuộc họp quản trị.
* Chia sẻ RIDA quan trọng (risks, impediments, dependencies, and assumptions) với các bên liên quan.
* Thảo luận các công việc cần ưu tiên, phạm vi, kinh phí và lịch trình.
* Đảm bảo rằng sản phẩm tồn đọng có thể nhìn thấy, minh bạch và rõ ràng.

### Scrum master

Scrum được xây dựng bởi một **Scrum master**. **Scrum master** không phải là trưởng nhóm truyền thống hay người quản lý dự án mà đóng vai trò như một người loại bỏ các yếu tố gây cản trở quá trình hoàn thành mục tiêu đề ra của nhóm, giúp quá trình thực hiện các mục tiêu của sản phẩm và phân phối sản phẩm được trơn tru. Một trong những vai trò của **Scum master** khác với Project manager là **Project manager** sau này có thể có trách nhiệm quản lý con người và người chủ quản lý sự cố thì không.

Các nhiệm vụ chính:

* Giúp **Product owner** theo dõi công việc tồn đọng của sản phẩm đảm bảo công việc được hiểu rõ để nhóm có thể liên tục đạt được tiến độ công việc.
* Giúp nhóm thấu hiểu các yêu cầu từ các bên liên quan.
* Truyền đạt các nguyên tắc Scrum cho các thành viên trong nhóm.
* Giúp nhóm tránh những yếu tố cản trở trong quá trình làm việc, cho dù nội bộ hay bên ngoài nhóm.
* Thúc đẩy tính tự tổ chức và nâng cao trách nhiệm của các thành viên trong nhóm.
* Tạo điều kiện cho các sự kiện của nhóm để các công việc thường xuyên được đảm bảo tiến độ.

### Developers

Các nhà phát triển là những người sẽ thực hiện tất cả các công việc cần thiết để gia tăng giá trị của sản phẩm sau mỗi Sprint. Bất kỳ ai đóng góp trong việc phát triển và hỗ trợ hệ thống hoặc sản phẩm có thể gọi là Developer, bao gồm các nhà nghiên cứu, kiến trúc sư, nhà thiết kế, chuyên gia dữ liệu, nhà thống kê, nhà phân tích, kỹ sư, lập trình viên và người kiểm tra, …

Nhóm được khuyến khích thảo luận trực tiếp với khách hàng hay các bên liên quan để cung cấp sự hiểu biết tối đa về các yếu tố kỹ thuật và phản hồi ngay lập tức.

## **Workflow**

**Scrum Sprint**

Sprint là một sự kiện sẽ được lặp đi lặp lại với mục tiêu được xác định trước khi bắt đầu, nhiệm vụ chính là giải quyết các công việc tồn đọng. Một Sprint có thể kéo dài từ 1 tuần đến 1 tháng nhưng phổ biến nhất sẽ là 2 tuần.

### **Sprint planning**

Tại thời điểm bắt đầu Sprint, nhóm sẽ tổ chức 1 buổi họp để:

* Thảo luận mục tiêu của Sprint, mô tả ngắn gọn những việc sẽ được hoàn thành khi Sprint kết thúc.
* Chọn các công việc đang tồn đọng liên quan đến mục tiêu đề ra của Sprint hay cần được ưu tiên ngay thời điểm hiện tại.
* Thời lượng tối đa của việc lập kế hoạch 1 Sprint diễn ra trong bốn tuần là tám giờ. [3] Khi công việc chi tiết được xác định, một số công việc tồn đọng có thể được chia nhỏ hoặc trả lại công việc tồn đọng sản phẩm nếu nhóm tin rằng họ không thể hoàn thành công việc đó trong một lần Sprint.

### **Daily scrum**

[1]:  *Schwaber, Ken; Sutherland, Jeff (November 2017),*[*The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*](https://www.scrumguides.org/scrum-guide.html)*, retrieved May 13, 2020*

[2] [*"Lessons learned: Using Scrum in non-technical teams"*](https://www.agilealliance.org/resources/experience-reports/lessons-learned-using-scrum-in-non-technical-teams/). Agile Alliance*. Retrieved April 8, 2019*.