Agile là gì? Scrum là gì? Các công cụ quản lý dự án theo Agile mà bạn nên biết

Published by

[**Nhat Nguyen**](https://topdev.vn/blog/author/nhat/)

17605

Có rất nhiều phương thức phát triển phần mềm theo quy chuẩn, và một trong số đó là phương thức phát triển phần mềm theo mô hình Scrum ([Scrum là gì](https://topdev.vn/blog/agile-la-gi-scrum-la-gi/#scrum-la-gi)?). Bài viết này sẽ giải thích các khái niệm cơ bản nhất cũng như những giá trị cốt lõi về Agile để bạn có thể nắm chắc được.

**Agile là gì?**

**Agile**là một phương pháp phát triển phần mềm linh hoạt, là một hướng tiếp cận cụ thể cho việc quản lý dự án phần mềm. Nó gồm một quá trình làm việc tương tác và tích hợp để có thể đưa sản phẩm đến tay người dùng càng nhanh càng tốt.

*Trong các dự án phần mềm, đặc biệt là các dự án chúng ta sẽ gặp rất nhiều khó khăn trong việc thu thập đầy đủ và chính xác các requirements của product để lập plan tốt ngay từ đầu. Có quá nhiều vấn đề gây ảnh hưởng đến việc phát triển phần mềm mà chúng ta không lường trước được. Ví dụ như những vấn đề có thể đến từ những yếu tố như kinh doanh, kỹ thuật, con người, thời gian ra mắt ….*

Xem tiếp...

Những phương pháp phát triển phần mềm theo cách truyền thống ngày càng bộc lộ nhiều nhược điểm và tỷ lệ các dự án thất bại cao trong thời kỳ bùng phát của ngành công nghệ. Nhận ra vấn đề đó, một số cá nhân và công ty riêng lẻ đã đưa ra các phương pháp phát triển phần mềm hiện đại hơn và khác nhau để thích ứng với tình hình mới.

Những phương thức phát triển phần mềm này giúp phần nào giải quyết được một số vấn đề nhưng lại phát sinh vấn đề khác về sự cộng tác, kỹ thuật, công cụ, hướng phát triển, chia sẻ ….

Vào năm 2001, **bản tuyển ngôn Agile (**[**Agile Manifesto**](https://agilemanifesto.org/)**)** đã được thống nhất và ra đời bởi một nhóm người có uy tính trong phát triển phần mềm:

* **Individuals and interactions over processes and tools**: Cá nhân và sự tương tác hơn là quy trình và công cụ
* **Working software over comprehensive documentation**: Phần mềm chạy tốt hơn là tài liệu đầy đủ
* **Customer collaboration over contract negotiation**: Cộng tác với khách hàng hơn là đàm phán hợp đồng
* **Responding to change over following a plan**: Phản hồi với sự thay đổi hơn là bám theo kế hoạch

**[Hoàn thiện và đo vận tốc nhóm Agile Scrum của bạn](https://topdev.vn/blog/?p=7961" \t "_blank)**

**1. Cá nhân và sự tương tác hơn là quy trình và công cụ**

Đặt trọng tâm vào con người và sự tương hỗ giữa những thành viên trong team. Nếu dự án có những thành viên có năng lực, chịu làm việc cùng nhau thì sẽ mang đến thành công cho dự án.

*Nếu dự án của bạn có quy trình làm việc tốt, được hỗ trợ bởi những công cụ tốt nhất nhưng những thành viên không thống nhất hoặc cùng nhìn về một hướng thì khả năng dự án thất bại là rất lớn. Nói điều này không có nghĩa là phủ nhận tầm quan trọng của quy trình và công cụ nhưng trong Agile nó được đặt sau yếu tố con người.*

Xem tiếp...

***Quy trình*** là các thủ tục cần thiết để phát triển dự án như thiết kế, sau đó đến lập trình, rồi kiểm tra QA/QC. Hay để đưa ra một chức năng nào đó cần phải có sự đồng ý của bộ phận QA/QC …. Quy trình này do mỗi công ty quy định và bắt  buộc các nhân viên khi tham gia vào dự án phải tuân thủ.

***Công cụ*** là phần mềm được sử dụng trong dự án như : Phần mềm quản lý công việc, phần mềm quản lý source code, phần mềm quản lý lỗi… Có rất nhiều công cụ được sử dụng để hỗ trợ một tổ chức vận hành.

**2. Phần mềm chạy tốt hơn là tài liệu đầy đủ**

Trong một số [quy trình phát triển phần mềm](https://topdev.vn/blog/quy-trinh-phat-trien-phan-mem/), việc tạo ra và cập nhật các tài liệu về sản phẩm là bắt buộc. Nhóm [lập trình viên](https://topdev.vn/blog/lap-trinh-vien/) không thể hoặc không đồng ý tiến hành công việc nếu không có tài liệu đặc tả về yêu cầu, thiết kế hệ thống.

Nhóm kiểm thử thì yêu cầu tài liệu về sản phẩm để có thể viết trường hợp kiểm thử và kiểm thử được. Nhóm QA đòi tất cả các tài liệu phải được viết trước khi sản phẩm được giao cho khách hàng nếu không thì không đủ điều kiện, chuẩn để giao sản phẩm cho khách hàng.

Việc viết tài liệu thật ra rất mất nhiều thời gian và được cho là rất chán. Ý tưởng ở đây là tại sao mình phải tập trung quá nhiều cho việc không cần thiết mà không dành thời gian đó để trao đổi để hiểu thêm về công việc phải làm. Sau đó đúc kết và chỉ viết những gì mà mọi người cần đọc.

**3. Cộng tác với khách hàng hơn là đàm phán hợp đồng**

Ta luôn nghe các câu này “Khách hàng là thượng đế” hay “khách hàng luôn luôn đúng”. Tuy nhiên thì khách hàng có nhiều dạng. Cách duy nhất để có thể làm việc tốt là phải cộng tác với khách hàng để hiểu được khách hàng muốn gì và cần gì để có thể tư vấn và điều chỉnh thay vì chỉ dựa vào những điều đã quy định trong hợp đồng.

Trao đổi và thảo luận với khách hàng về sự cần thiết có hay không của một chức năng trong sản phẩm, từ đó quyết định là có nên làm hay không. Tất nhiên để thuyết phục khách hàng thì cần có số liệu nghiên cứu cụ thể chẳng hạn.

**4. Phản hồi với sự thay đổi hơn là bám theo kế hoạch**

Có một điểm chung là hầu hết những dự án đều có sự thay đổi điều chỉnh khi triển khai. Sự thay đổi đó có thể là thay đổi về requirements, thay đổi tech stack, thay đổi nhân sự, thay đổi deadline, thay đổi phương thức làm việc… mặc dù kế hoạch đã được định ra rõ ràng từ đầu.

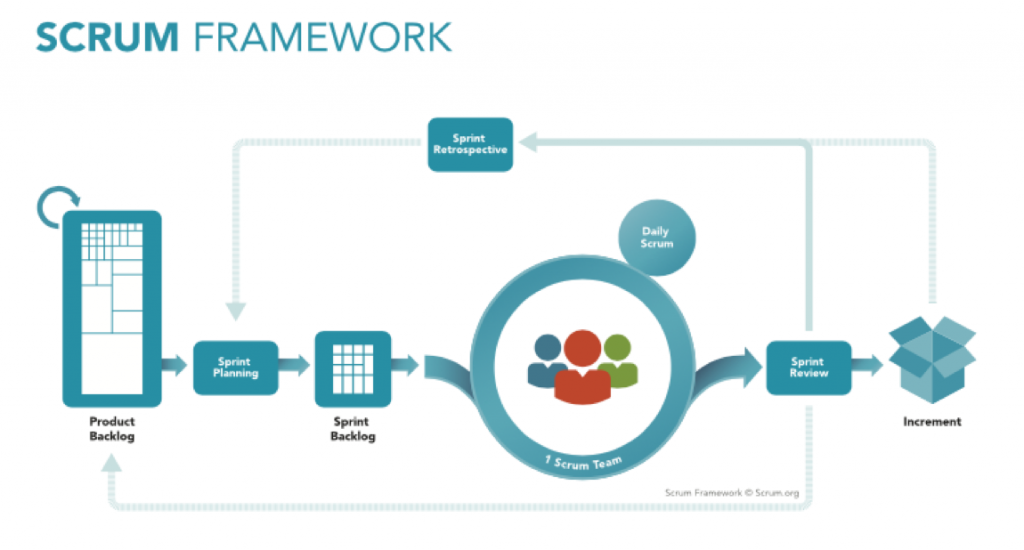
Agile không khuyến khích cho sự thay đổi nhưng khuyến khích chúng ta tập thích nghi với thay đổi.

Một trong các nguyên tắc cơ bản của agile là “phần mềm chạy tốt chính là thước đo của tiến độ”. Nguyên tắc này giúp nhóm dám loại bỏ đi các công việc dư thừa không trực tiếp mang lại giá trị cho sản phẩm.

Để vận hành được cơ chế “làm việc dựa trên giá trị”, nhóm agile thường làm việc trực tiếp và thường xuyên với khách hàng, cộng tác trực tiếp với họ để biết yêu cầu nào có ***độ ưu tiên cao hơn***, mang lại giá trị hơn sớm nhất có thể cho dự án. Nhờ đó các dự án agile thường giúp khách hàng tối ưu hóa được giá trị của dự án. Một cách gần như trực tiếp, agile gia tăng đáng kể độ hài lòng của khách hàng.

**Scrum là gì?**

**Scrum** là một “bộ khung làm việc” cơ bản để tiếp cận những công việc phức tạp. Dựa trên bộ khung này, nhóm làm việc có thể áp dụng những quy trình, kỹ thuật khác nhau cho công việc của mình… Nó là một thành viên của họ [Agile](https://topdev.vn/blog/agile-la-gi/).

Credit: Scrum.org

**Scrum có ích gì cho phát triển phầm mềm hiện nay**

Nó giúp loại bỏ những công đoạn phức tạp và chỉ tập trung vào những công đoạn cần thiết đáp ứng được nhu cầu của khác hàng đưa ra. Ba yếu tố nòng cốt tạo thành một mô hình quản lý tiến trình thực nghiệm gồm: **sự minh bạch** (transparency), **thanh tra** (inspection) và **thích nghi** (adaptation).

**Ba giá trị cốt lõi của Scrum**

**1. Minh bạch**

Muốn áp dụng thành công Scrum, các thông tin liên quan đến quá trình phải mình bạch và thông suốt. Các thông tin có thể là tầm nhìn của sản phẩm, yêu cầu của khách hàng, tiến độ công việc, các rào cản khác…

Từ đó mọi thành viên ở vai trò khác nhau có đầy đủ thông tin cần có để tiến hành quyết định trong việc nâng cao hiệu quả công việc.

**2. Thanh tra**

Phải thường xuyên thanh tra các hoạt động trong Scrum và tiến độ đến đích để phát hiện các bất thường không theo ý muốn. Tần suất thanh tra không nên quá dày để khỏi ảnh hưởng đến công việc. Công tác thanh tra khi được thực hiện bởi người có kĩ năng tại các điểm quan trọng của công việc sẽ giúp cải tiến liên tục trong Scrum.

**3. Thích nghi**

Scrum mang lợi thế là tính linh hoạt rất cao, nhờ đó mang lại tính thích nghi cao. Dựa vào thông tin liên tục và minh bạch từ quá trình thanh tra và làm việc, Scrum có thể cho lại các thay đổi tích cực, nhờ đó mang lại thành công cho dự án.

**Lợi ích mà Scrum mang lại**

Tính minh bạch, kiểm tra, và thích nghi là 3 nền tảng cơ bản của Scrum. Và dưới đây là những lý do tại sao nên dùng Scrum.

1. Cải thiện chất lượng phần mềm, dễ học và dễ sử dụng.
2. Rút ngắn thời gian phát hành phần mềm, cho phép khách hàng sử dụng sản phẩm sớm hơn.
3. Nâng cao tinh thần đồng đội, tối ưu hóa hiệu quả và nỗ lực của đội phát triển.
4. Gia tăng tỷ suất hoàn vốn đầu tư (ROI)
5. Tăng mức độ hài lòng của khách hàng
6. Kiểm soát dự án tốt, cải tiến liên tục
7. Giảm thiểu rủi ro khi xây dựng sản phẩm

**Các khái niệm cơ bản Scrum**

**1. Scrum Team**

Scrum team chia làm 3 vai trò bao gồm những thành phần sau:

* **Product Owner**: Nhiệm vụ của Product Owner là đảm bảo việc quản lý những công việc còn tồn đọng (**Product backlog**) của việc phát triển sản phẩm phần mềm. Product Owner phải liên tục cập nhật thông tin cho các thành viên trong team để họ hiểu về yêu cầu hay các tính năng cần có của sản phẩm ngay cả khi họ không trực tiếp phát triển tính năng đó.
* **Development Team**: là những [lập trình viên](https://topdev.vn/blog/lap-trinh-vien/) sẽ tham gia vào việc phát triển từng tính năng cụ thể. Các lập trình viên này có thể sẽ có kỹ năng khác nhau và một số sẽ giỏi về những kỹ năng nhất định. Tuy nhiên khi sử dụng Scrum thì tất cả các thành viên của Development Team yêu cầu phải có khả năng làm việc thay thế vị trí của nhau và không ai chỉ chịu trách nhiệm phát triển một (hoặc một số) tính năng nhất định.
* **Scrum Master**: sẽ chịu trách nhiệm cho việc lên kế hoạch để phân công công việc, sắp xếp thứ tự ưu tiên giải quyết những công việc tồn đọng nào có trong **Backlog** trước, tổ chức các buổi họp với Product Owner để theo dõi tình hình và nắm thông tin cần thiết.

**2. Sprint**

**Sprint**là mộ phân đoạn lặp đi lặp lại trong quy trình phát triển phần mềm, có khung thời gian thường là 1 tháng (từ 1 – 4 tuần) mà theo đó sản phẩm sẽ được *release* phiên bản mới. Khi một Sprint kết thúc thì Scrum Master cần phải chuyển trạng thái của nó sang *Done*.

Khi bắt đầu một Sprint thì Scrum Master cần đưa ra mục tiêu của Sprint đó và mục tiêu này không được phép thay đổi cho tới khi Sprint hoàn thành. Tuy nhiên Product Owner vẫn có quyền huỷ một Sprint trước thời hạn kết thúc của nó.

Mặc dù để làm điều này thì Product Owner cần sự đồng thuận của Development Team cũng như Scrum Master. Sau khi một Sprint kết thúc thì các bên sẽ dựa trên kết quả của Sprint đó để lên kế hoạch cho Sprint tiếp theo.

**3. Sprint Planning**

Đây là bước đầu tiên cần phải thực hiện trước khi một Sprint bắt đầu. Development team họp với Product Owner để lên kế hoạch cho một sprint. Những công việc nào cần phải được hoàn thành trong Sprint này và làm sao để có thể hoàn thành những công việc này.

Sau khi thống nhất được số lượng công việc, thời gian hoàn thành thì chúng ta có thể bắt đầu Sprint. Trong khi thực hiện một Sprint chúng ta sẽ phải có những buổi họp được gọi là Daily Sprint hay Daily Meeting.

**4. Daily Sprint**

Các buổi họp Daily Sprint thường kéo dài khoản 15 phút, trong buổi họp này tất cả các thành viên sẽ lần lượt báo cáo lại:

* Những gì họ đã làm được ngày hôm qua
* Những gì họ cần làm ngày hôm nay
* Những khó khăn mà họ gặp phải

Mỗi buổi họp này sẽ giúp việc dự kiến được kế hoạch đưa ra trong Sprint đang làm sẽ tiến triển ra sao và liệu có cần phải cập nhật lại bản kế hoạch đã đưa ra hay không. Tất nhiên cần nhớ rằng việc thay đổi kế hoạch này không bao gồm thay đổi mục tiêu đã đưa ra của Sprint.

Ví dụ bạn có thể tăng thêm thời gian để hoàn thành một chức năng và qua đó khiến Sprint phải kéo dài hơn dự kiến. Tuy nhiên mục tiêu của Sprint là cho phát hành một phiên bản mới cần được giữ nguyên.

**5. Sprint Review**

Là công việc được thực hiện bởi nhóm phát triển và product owner ở cuối mối Sprint nhằm đánh giá lại kết quả thực hiện được. Từ lúc Sprint mới hoàn thành và qua đó đưa ra những chỉnh sửa, thay đổi cần thiết ở Sprint sau.

**6. Sprint Restrospective**

Dưới sự trợ giúp của Scrum master, team phát triển sẽ tổng kết những kiến nghị và đánh giá từ bước Sprint Review ở trên để đưa ra những cải tiến nhằm nâng cao hiệu quả làm việc cũng như sản phẩm.

**[Những điểm khác biệt giữa Kanban và Scrum](https://topdev.vn/blog/?p=869" \t "_blank)**

**[Những điểm khác biệt giữa Kanban và Scrum](https://topdev.vn/blog/?p=869" \t "_blank)**

**7. Các công cụ (artifacts) Scrum**

Scrum sử dụng các công cụ rất đơn giản nhưng hiệu quả để trợ giúp công việc.

**Product backlog**

Đây là danh sách ưu tiên các tính năng (feature) hoặc đầu ra khác của dự án. Có thể hiểu như là danh sách yêu cầu (requirement) của dự án.

Product Owner chịu trách nhiệm sắp xếp độ ưu tiên cho từng hạng mục (Product Backlog Item) trong Product Backlog dựa trên các giá trị do Product Owner định nghĩa (thường là giá trị thương mại – business value).

**Sprint backlog**

Đây là bản kế hoạch cho một Sprint; là kết quả của buổi họp lập kế hoạch (Sprint Planning).

Với sự kết hợp của Product Owner, nhóm sẽ phân tích các yêu cầu theo độ ưu tiên từ cao xuống thấp để hiện thực hóa các hạng mục trong Product Backlog dưới dạng danh sách công việc (TODO list).

**Burndown Chart**

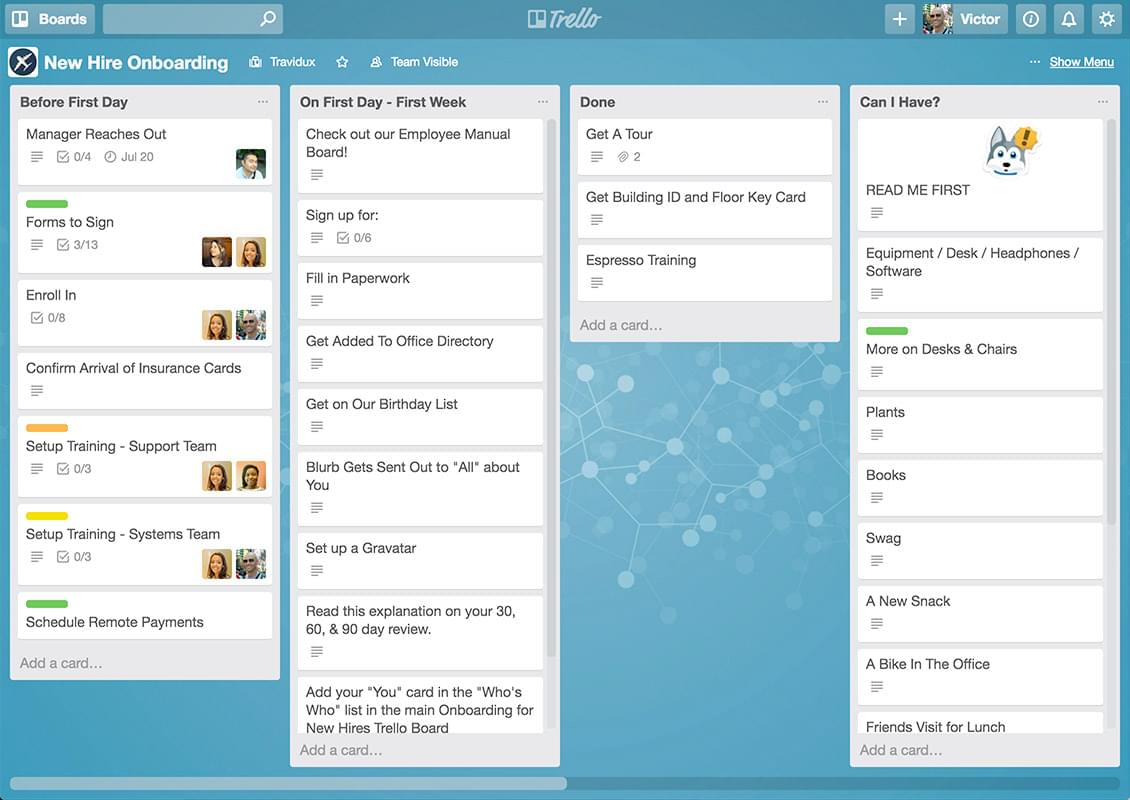
Đây là biểu đồ hiển thị xu hướng của dự án dựa trên lượng thời gian cần thiết còn lại để hoàn tất công việc.

Burndown Chart có thể được dùng để theo dõi tiến độ của Sprint (được gọi là Sprint Burndown Chart) hoặc của cả dự án (Project Burndown Chart).

Biểu đồ burndown không phải là một thành tố tiêu chuẩn của Scrum theo định nghĩa mới, nhưng vẫn được sử dụng rộng rãi do tính hữu ích của nó.

**Các công cụ quản lý dự án theo Agile mà bạn nên biết**

[**Trello**](https://trello.com/)

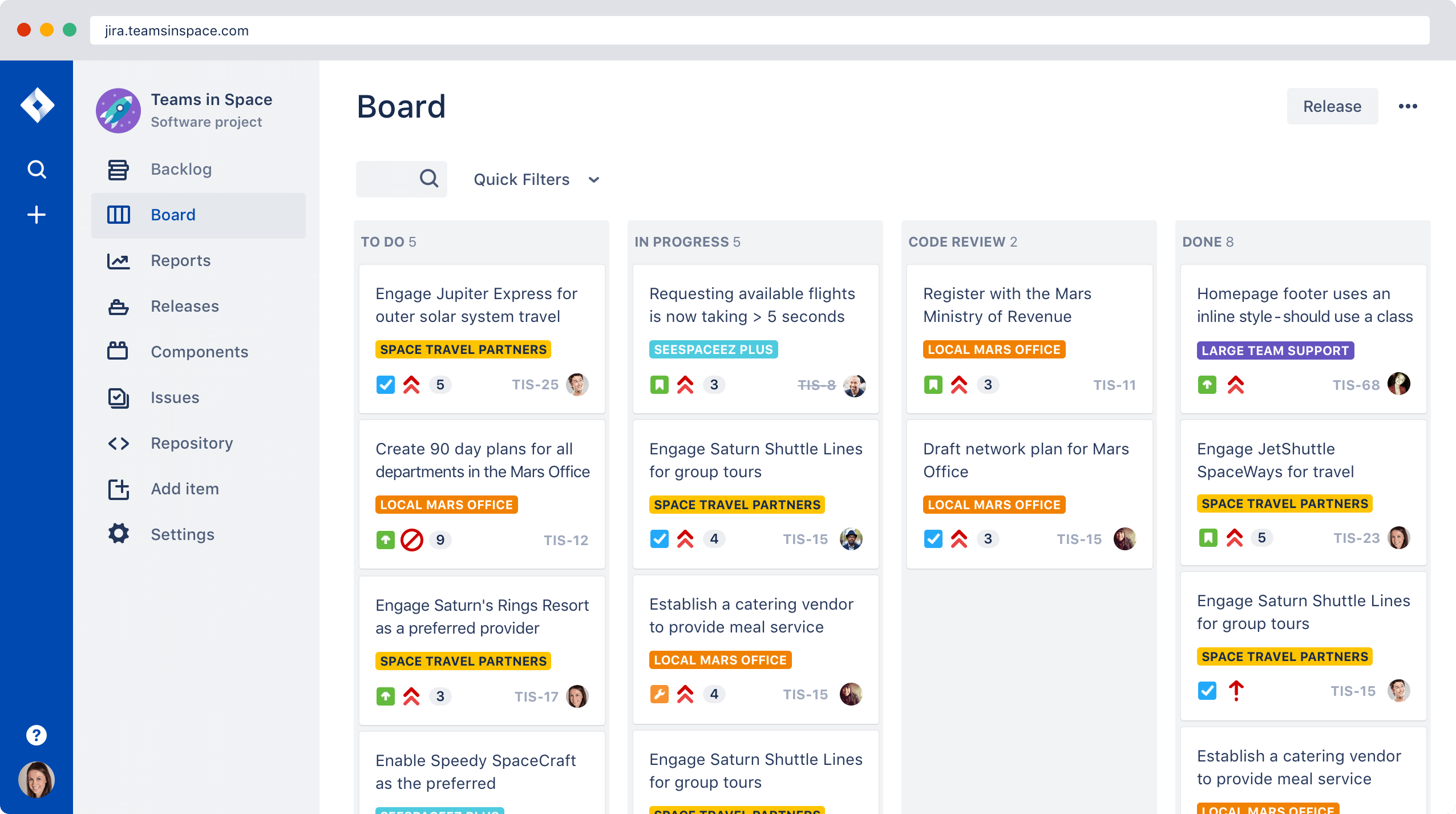


Đây là một trong những ứng dụng quản lý dự án nổi tiếng và được sử dụng nhiều nhất. Nó có cả tài khoản miễn phí và cao cấp mang đến cho bạn cơ hội tuyệt vời để sử dụng hầu hết các chức năng phổ biến.

Cấu trúc của Trello dựa trên phương pháp kanban. Tất cả các dự án được đại diện bởi các bảng, có chứa danh sách. Mọi danh sách đều có các thẻ lũy tiến mà bạn được tạo dưới dạng kéo và thả. Người dùng có liên quan đến bảng, có thể được gán cho thẻ.

Tóm lại, nó có nhiều tính năng hay, nhỏ nhưng không kém phần hữu ích: viết bình luận, chèn tệp đính kèm, ghi chú, ngày đáo hạn, danh sách kiểm tra, nhãn màu, tích hợp với các ứng dụng khác, v.v. Ngoài ra, Trello được hỗ trợ bởi tất cả các nền tảng di động. Trello là công cụ có thể được sử dụng cho cả công việc và các quy trình cá nhân.

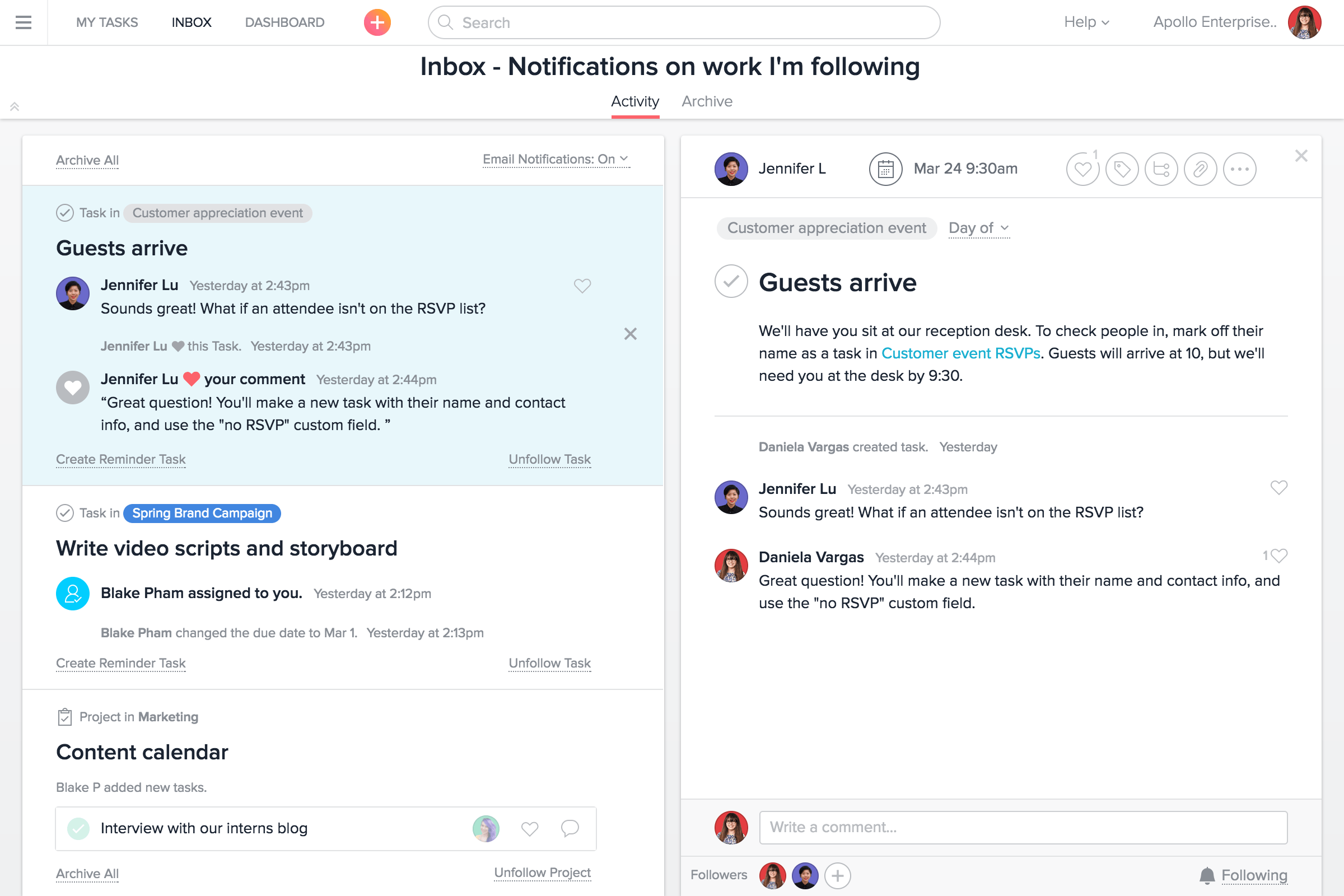
[**JIRA**](https://jira.atlassian.com/secure/Dashboard.jspa)



JIRA là một công cụ được phát triển để theo dõi lỗi, theo dõi vấn đề và quản lý dự án cho các quy trình phát triển phần mềm và di động. Bảng điều khiển JIRA có nhiều chức năng & tính năng hữu ích có thể xử lý các vấn đề khác nhau một cách dễ dàng.

Một số tính năng và sự cố chính: loại sự cố, quy trình làm việc, màn hình, trường, thuộc tính vấn đề. Một số tính năng bạn sẽ không tìm thấy ở nơi khác. Bảng điều khiển trên JIRA có thể được tùy chỉnh để phù hợp với quy trình kinh doanh của bạn.

[**Asana**](https://asana.com/)



Asana là công cụ quản lý công việc cho phép các nhóm chia sẻ, lập kế hoạch, tổ chức và theo dõi tiến trình của các nhiệm vụ mà mỗi thành viên đang thực hiện. Nó đơn giản, dễ sử dụng và miễn phí cho tối đa 30 người dùng trong một nhóm.

Như tất cả các nền tảng phần mềm quản lý dự án Agile trước đây với mục tiêu chính là cho phép quản lý các dự án và nhiệm vụ. Điều đáng chú ý là bạn không cần phải có email để sử dụng Asana. Mỗi nhóm có thể tạo nơi làm việc sẽ chứa các dự án và nhiệm vụ của dự án: mỗi tác vụ có thể có ghi chú, nhận xét, tệp đính kèm và thẻ.

Công cụ này có thể được sử dụng cho các quy trình nhỏ và cho các quy trình lớn mà không có bất kỳ giới hạn nào trong các ngành hoặc bộ phận.

**Resource cho bạn tìm hiểu về Agile và Scrum:**

* [Việc làm Agile](https://topdev.vn/viec-lam-it?q=Product+Manager): Có rất nhiều tại TopDev với mức lương cực hấp dẫn.
* [Scrum.org](https://www.scrum.org/): đầy đủ kiến thức cơ bản, nâng cao về Scrum và các chứng chỉ Scrum.
* [Agile Manifesto](http://agilemanifesto.org/): cơ bản về Agile, tuyên ngôn Agile cho người mới bắt đầu.
* [Agile Vietnam Group](https://www.facebook.com/groups/agilevietnam/) và [Agile forum Vietnam](http://www.agilevietnam.org/): diễn đàn lớn nhất về Agile tại Việt Nam, cùng chia sẻ thông tin, kiến thức, sự kiện về Agile.

**Tổng kết**

Bạn cần lưu ý rõ điều này Agile không phải là “một phương pháp” mà là tư duy, cách tiếp cận, là tập hợp những phương pháp, sự thực hành dựa trên những giá trị và nguyên tắc nêu ra trong bản tuyên ngôn Agile.