

Quản lý dự án Công Nghệ Thông Tin

Phần 3. Các phương Pháp quản lý dự án

Lịch sử các phương pháp quản lý dự án

Giai Đoạn Đầu: Thời kỳ Trước 1970

- **Sự Khởi Đầu:** Quản lý dự án trong CNTT bắt đầu hình thành, chủ yếu xoay quanh phát triển hệ thống thông tin lớn, thường sử dụng các phương pháp truyền thống và tuyến tính.
- **Phương pháp Waterfall:** Được ưa chuộng trong giai đoạn này, mô hình Waterfall tập trung vào việc hoàn thành từng giai đoạn của dự án theo thứ tự cố định từ phân tích yêu cầu đến thiết kế, triển khai, và bảo trì.

Thập Niên 1970 - 1980: Sự Phát Triển và Thích Ứng

- **Phức tạp hóa của Dự Án:** Các dự án CNTT trở nên phức tạp hơn với sự gia tăng của công nghệ và kỳ vọng của người dùng.
- **Ra đời các Phương pháp Mới:** Phương pháp như PERT và CPM (Critical Path Method) bắt đầu được sử dụng để quản lý thời gian và nguồn lực hiệu quả hơn trong các dự án CNTT.

Lịch sử các phương pháp quản lý dự án

Thập Niên 1990: Sự Xuất Hiện của Agile

- **Sự Cần thiết cho Sự Linh hoạt:** Với sự thay đổi nhanh chóng của công nghệ, các phương pháp truyền thống bắt đầu được xem xét lại. Cần có cách tiếp cận mới linh hoạt hơn.
- **Agile Manifesto (2001):** Sự ra đời của Agile Manifesto đánh dấu một bước ngoặt, đề cao sự linh hoạt, tương tác, và thích ứng nhanh với thay đổi.

Thập Niên 2000: Sự Phổ Biến của Agile và Scrum

- **Agile và Scrum:** Phương pháp Scrum, một phần của Agile, trở thành một cách tiếp cận phổ biến trong quản lý dự án phần mềm do khả năng thích ứng với thay đổi và tập trung vào hợp tác nhóm.
- **Kanban và Lean:** Các phương pháp linh hoạt khác như Kanban và Lean bắt đầu được áp dụng rộng rãi.

Lịch sử các phương pháp quản lý dự án

Hiện Đại: Sự Đa Dạng và Tích Hợp Công Nghệ

- **Kỷ nguyên Kỹ thuật số:** Với sự phát triển của đám mây, big data, và AI, quản lý dự án CNTT bước vào kỷ nguyên kỹ thuật số, đòi hỏi cách tiếp cận tổng hợp và tích hợp công nghệ.
- **DevOps và Agile Mở rộng:** Sự hội nhập giữa phát triển (Dev) và vận hành (Ops) cùng với sự mở rộng của Agile ra ngoài phần mềm, như trong quản lý sản phẩm và dự án tổ chức.

Tương lai: AI làm PM

Các phương pháp quản lý dự án phổ biến hiện nay

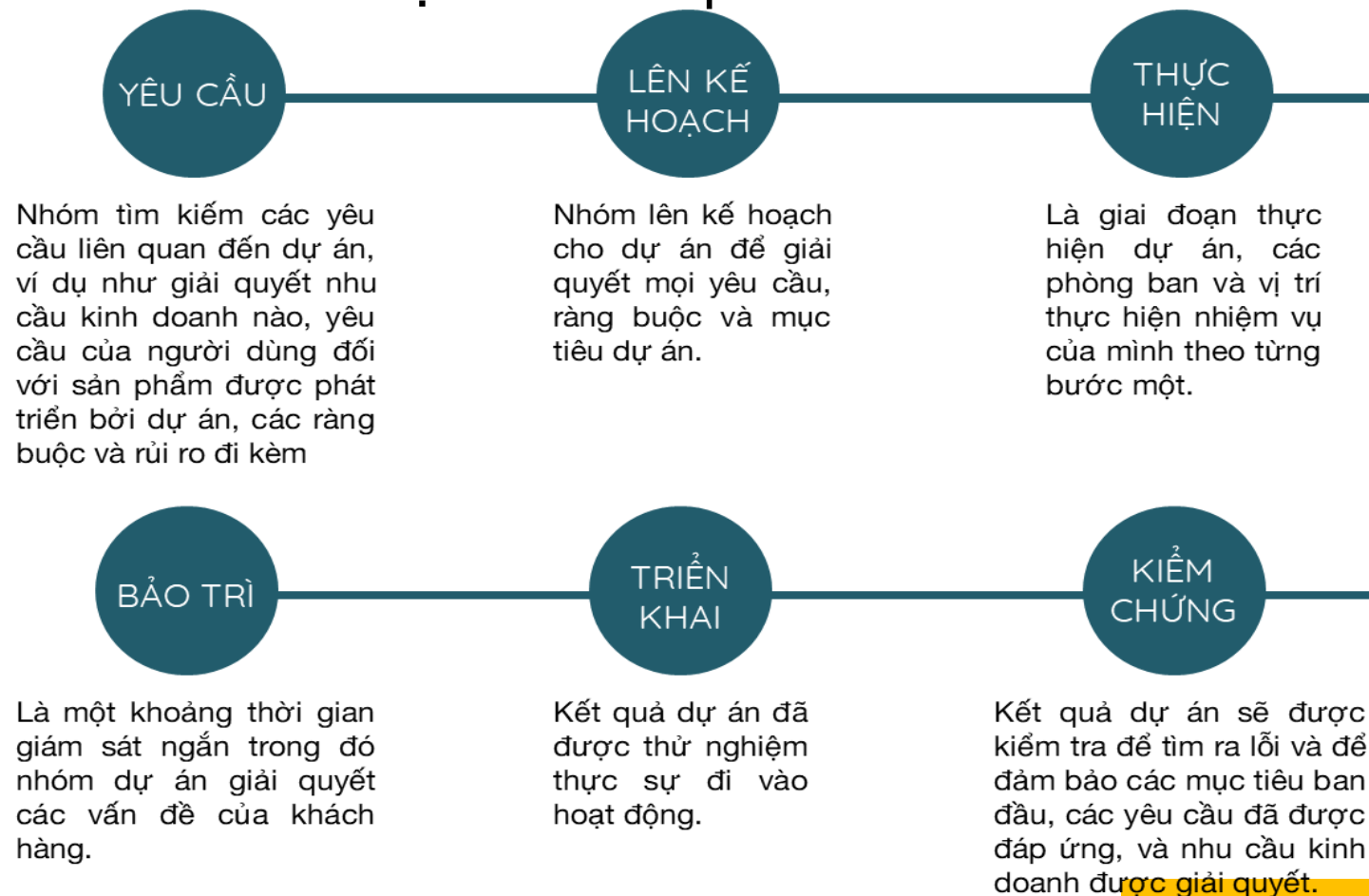
- Có thể thống kê 6 phương pháp Quản lý dự án phổ biến nhất hiện nay:

1. Waterfall
2. Kanban
3. Agile
4. Extreme
5. Lean
6. Scrum

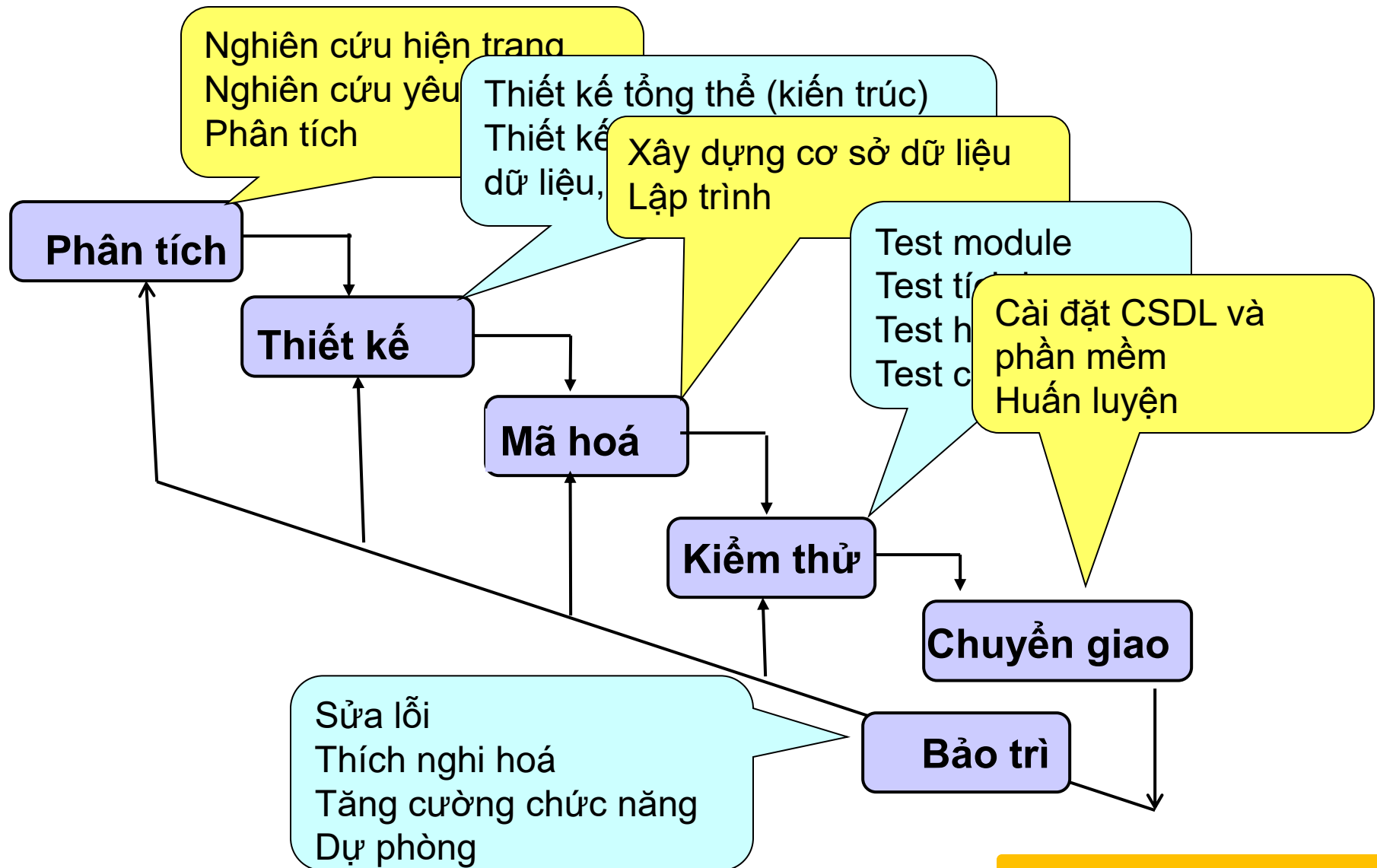


1. WATERFALL - Mô hình thác nước

- Là một trong những mô hình quản lí dự án dễ hiểu và dễ quản lí nhất hiện nay,
- Mô hình thác nước là phương pháp quản lí dự án dựa trên qui trình thiết kế tuần tự và liên tiếp.



1. WATERFALL - Mô hình thác nước



1. WATERFALL - Mô hình thác nước

● Ưu điểm của Waterfall:

- Waterfall rất tuyệt vời nếu bạn có thể đưa ra một kế hoạch, quy trình chặt chẽ và những dự trù phát sinh ngay từ đầu.
- Là phương pháp đơn giản, dễ sử dụng và dễ hiểu.
- Quy trình thực hiện rõ ràng, phân phối dự án dễ và nhanh, dễ dàng phân bổ chi phí hợp lý.
- Phù hợp với dự án nhỏ và không phát sinh nhiều yêu cầu mới trong quá trình triển khai
- Dễ dàng quản lý và theo dõi tiến độ dự án đang đi đến đâu do mỗi giai đoạn đều có quá trình cụ thể, danh sách nhiệm vụ rõ ràng và kết quả nằm trong tầm dự đoán.

1. WATERFALL - Mô hình thác nước

● Nhược điểm của Waterfall:

- Thích ứng kém
- Bỏ qua phản hồi người dùng ở các giai đoạn sau
- Thử nghiệm chỉ bắt đầu khi quá trình thực hiện kết thúc

● Waterfall phù hợp với dự án như thế nào?

Việc áp dụng mô hình Waterfall được khuyến khích khi người thực hiện nắm rõ yêu cầu của dự án tốt nhất, đòi hỏi về tính rõ ràng và tính ổn định cao như:

- Khi mà đội dự án đã có kinh nghiệm làm việc, trình độ chuyên môn và kỹ thuật cao bởi mô hình này đòi hỏi sự chính xác ngay từ đầu.
- Khách hàng xác định được yêu cầu cụ thể, chính xác ngay từ đầu và ít có khả năng thay đổi, loại bỏ được những yêu cầu mập mờ, không rõ ràng.
- Đối với những khách hàng lớn mà phong cách làm việc của họ chủ yếu theo mô hình truyền thống hoặc những khách hàng lo ngại có nhiều thay đổi trong dự án.
- Nắm vững được công nghệ và sự phát triển của công nghệ.

2. KANBAN - Bảng và thẻ

- Theo dịch nghĩa tiếng Nhật, Kanban là có nghĩa là thẻ thị giác (với từ kan là thị giác và từ ban là thẻ).
- Bảng Kanban như một công cụ trực quan, nơi bạn có thể thấy nhìn thấy, bằng mắt, vị trí của từng tác vụ trong quy trình công việc.
- Bảng được chia thành nhiều cột cho từng giai đoạn trong quy trình. Các cột này có thể được tùy chỉnh theo ý thích của bạn, tùy thuộc vào dự án bạn đang làm việc.

Bảng Kanban

Danh sách Kanban

Thẻ Kanban



2. KANBAN - Bảng và thẻ

Bảng Kanban

Danh sách Kanban

Thẻ Kanban



- **Bảng Kanban:** Là bảng chứa những việc cần làm của một dự án hoặc quy trình làm việc, trong công cụ quản lý truyền thống thì đây là “dự án” hoặc “không gian làm việc”.
- **Danh sách Kanban:** Một danh sách chứa một tập hợp các thẻ, thường là những thẻ trong cùng một giai đoạn của quy trình; công cụ quản lý dự án truyền thống gọi đây là "danh sách việc cần làm" hoặc "danh sách công việc".
- **Thẻ Kanban:** Thẻ chứa nội dung liên quan đến bảng và danh sách, chẳng hạn như một nhiệm vụ phải hoàn thành hoặc một sản phẩm sẽ được thực hiện; thẻ Kanban công cụ quản lý dự án truyền thống gọi là "việc cần làm" hoặc "nhiệm vụ".

📋 Danh sách công việc

Table List Board Member Ngày tạo Hiển thị: Tất cả + Thêm

Chưa phân loại 5

- Chỉnh lại site Luxstay ☐
- Đổi ngôn ngữ Anh sang Việt site attovietnam.talent.vn ☐
- Triển khai website Sacombank ☐ 2
- Cố định 2 jobs đã ghim trong hiring trên careersite của... ☐
- Triển khai website Hecatech ☐
- Đổi ngôn ngữ site minhhaigroup.talent.vn sang... ☐
- Sửa lại thông tin site của Phong Vũ ☐
- Đổi lại banner và icon cho site ACB ☐
- Sửa lại text trên trang elecom.talent.vn ☐
- Triển khai Site ACB ☐
- Triển khai website ATZ LIFE ☐
- Triển khai site ACB ☐
- Triển khai site Phong Vu ☐

To do 25

- Thay CV mẫu trên career site của KH Phú Mỹ Group ☐ Long 1 17/05
- Chỉnh sửa site Truemoney ☐ Long 1 03/05
- Triển khai Website PTF ☐ Long 1
- Chỉnh sửa website Pizza4p ☐ Long 1 03/04
- Thay file webiste pizzahut.talent.vn ☐ Long 2 02/04
- Triển khai website QSR Việt Nam ☐ Long 03/07
- Chỉnh sửa Banner Web Scommerce ☐ Phương 4 31/07

Doing 8

- Thay đổi thông tin site Decathlon ☐ Long 02/08
- Chỉnh sửa site bamboo airways ☐ Phương 1 07/01
- Thay banner trên PC + Mobile site PTF ☐ Long 30/07
- Chỉnh sửa thông tin site Đồng Tâm ☐ Phương 1 17/07
- Gắn GA cho site Sacombank ☐ Long 1 25/06
- chỉnh sửa site Mcdonald's ☐ Long 17/06
- Chỉnh sửa nội dung website PTF ☐ Phương 1 17/06
- Chỉnh sửa website McDonald's ☐ Long 01/06
- Chỉnh sửa web PTF ☐

Done! 195

- Tạo website Mc Donald's ☐ Phương 1 28/09
- Tạo web cho Vietnam ☒ Phương
- Chỉnh sửa giao diện site tuyển dụng L&A ☒ Phương 1
- Chỉnh sửa giao diện site tuyển dụng L&A ☒ Phương 2 11/07
- Tạo web cho Lotus Ocean ☒

Rejected 1

- Triển khai site Eway ☐ Long 31/07
- Triển khai website Sotamo ☐ Long 2 10/05
- Thay file webiste pizzahut.talent.vn ☐ Long 2 02/04
- Chỉnh sửa website QSR Việt Nam ☐ Long 10/07
- Chỉnh sửa site ATZ life ☐ Long 02/07
- Chỉnh sửa giao diện website SCB ☐ Long 1 31/03

2. KANBAN - Bảng và thẻ

- **Ưu điểm của Kanban:**

- Mọi người đều ở trên cùng một mặt phẳng.
- Giảm trực tiếp chi phí và lãng phí.
- Kanban tiết lộ những vướng mắc trong quy trình làm việc.
- Hệ thống đơn giản, dễ hiểu
- Hệ thống phản ứng nhanh nhạy
- Yêu cầu tập trung vào các nhiệm vụ hiện tại cho đến khi hoàn thành

2. KANBAN - Bảng và thẻ

● **Nhược điểm của Kanban:**

- Kanban thường tập trung vào các tác vụ hàng ngày, nó thúc đẩy việc thực thi nhiệm vụ, nhưng có thể gây rủi ro cho bức tranh lớn hơn là chiến lược và những kết quả quan trọng nhất. Các nhiệm vụ nhỏ được hoàn thành, nhưng kết quả cuối cùng chưa chắc đã đạt được.
- Không có khung thời gian của từng giai đoạn
- Phải cập nhật bảng
- Kanban sẽ trở nên rất khó áp dụng nếu có quá nhiều hoạt động hoặc nhiệm vụ liên quan đến nhau trong một hệ thống.
- Đầu ra có thể không đảm bảo chất lượng

2. KANBAN - Bảng và thẻ

● Kanban phù hợp với dự án nào:

- Cần một hệ thống linh hoạt để thêm hoặc xóa các mục khi trong quá trình làm việc mà có thay đổi trong thời gian ngắn.
- Dự án của bạn nhấn mạnh nhiều về quy trình làm việc liên tục hơn là những thời hạn hoàn thành đơn lẻ và quan trọng.
- Không có nhiều áp lực về thời gian hoàn thành.
- Cần cải tiến liên tục trong quá trình làm việc.
- Bạn muốn nhóm có khả năng báo cáo kết quả bất cứ lúc nào.
- Nhóm của bạn thích cải thiện gia tăng các quy trình hiện có hơn là áp đặt một hệ thống mới triệt để.
- Hệ thống hiện tại dễ hiểu để áp dụng Kanban.
- Ưu tiên hàng đầu là đáp ứng nhu cầu của khách hàng.
- Đang gặp phải tồn đọng do công việc bị đình trệ, quy trình cơ bản đã ổn định nhưng hoạt động cần mượt mà và hiệu quả hơn.

2. KANBAN - Bảng và thẻ



3. AGILE - Dự án linh hoạt

- Khái niệm Agile (Agile Software Development) có nghĩa là phương thức phát triển phần mềm linh hoạt, được ứng dụng trong quy trình phát triển phần mềm với mục tiêu là đưa sản phẩm đến tay người dùng càng nhanh càng tốt.
- Agile giống như một phương pháp luận, một triết lý dựa trên hơn nguyên tắc phân đoạn vòng lặp (iterative) và tăng trưởng (incremental).
- Ngày nay Agile trở thành một phương thức quản lý dự án phổ biến nhất hiện nay với nhiều đại diện được gọi là các phương pháp “họ Agile”, được áp dụng trong rất nhiều lĩnh vực:
 - cách thức làm việc,
 - quản lí,
 - sản xuất,
 - dịch vụ,
 - sales,
 - marketing,
 - giáo dục...

2. AGILE – Tuyên ngôn Agile

Tuyên ngôn AGILE



Cá nhân và tương tác

hơn là

Quy trình và công cụ



Cộng tác với khách hàng

hơn là

Đàm phán hợp đồng



Phần mềm chạy tốt

hơn là

Tài liệu đầy đủ



Phản hồi với thay đổi

hơn là

Bám sát kế hoạch

2. AGILE – Tuyên ngôn Agile

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

3. AGILE – 12 nguyên lý

1. Ưu tiên cao nhất của là làm **hài lòng khách hàng** thông qua việc **phân phối sớm** và liên tục các phần mềm có giá trị.
2. **Hoan nghênh các yêu cầu thay đổi**, ngay cả trong giai đoạn muộn của việc phát triển. Các quy trình nhanh khai thác sự thay đổi vì lợi thế cạnh tranh của khách hàng.
3. **Cung cấp sản phẩm phần mềm thường xuyên**, từ vài tuần đến vài tháng, ưu tiên khoảng thời gian ngắn hơn.
4. Người kinh doanh và nhà phát triển phải **làm việc cùng nhau** hàng ngày trong suốt dự án.
5. Xây dựng các dự án xung quanh những **cá nhân có động lực**. Cung cấp cho họ môi trường và sự hỗ trợ mà họ cần, và tin tưởng để họ hoàn thành công việc.
6. Phương pháp hiệu quả nhất để truyền tải thông tin đến và trong nhóm phát triển là **trò chuyện trực tiếp**.

3. AGILE – 12 nguyên lý

7. Phần mềm làm việc là thước đo chính của **sự tiến bộ**.
8. Các quy trình Agile thúc **đẩy sự phát triển** bền vững. Các nhà tài trợ, nhà phát triển và người dùng sẽ có thể duy trì một tốc độ phát triển liên tục.
9. Sự **quan tâm** liên tục, sự xuất sắc về kỹ thuật và thiết kế tốt giúp tăng cường sự nhanh nhẹn.
10. Sự **đơn giản** - nghệ thuật tối đa hóa khối lượng công việc chưa hoàn thành - là điều cần thiết.
11. Các **kiến trúc, yêu cầu và thiết kế** tốt nhất **xuất hiện từ** các **nhóm dự án**.
12. Theo định kỳ, các nhóm **phản ánh** về cách trở nên hiệu quả hơn, sau đó **điều chỉnh** hoạt động của mình sao cho phù hợp.

3. AGILE – Các nguyên lý của Agile

12 AGILE PRINCIPLES

- 01 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- 02 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- 03 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- 04 Business people and developers must work together daily throughout the project.
- 05 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- 06 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- 07 Working software is the primary measure of progress.
- 08 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- 09 Continuous attention to technical excellence and good design enhances agility.
- 10 Simplicity – the art of maximizing the amount of work not done – is essential.
- 11 The best architectures, requirements, and designs emerge from self-organizing teams.
- 12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

3. AGILE - Dự án linh hoạt

- **Ưu điểm của Agile :**

- Thực hiện thay đổi dễ dàng
- Không cần phải nắm mọi thông tin ngay từ đầu
- Bàn giao nhanh hơn
- Chú ý đến phản hồi của khách hàng và người dùng
- Cải tiến liên tục

- **Nhược điểm của Agile :**

- Khó lên kế hoạch dự án
- Bắt buộc phải hướng dẫn và đào tạo chi tiết
- Ít tài liệu hướng dẫn
- Bắt buộc phải hợp tác để dự án thành công
- Chi phí cao

3. AGILE - Dự án linh hoạt

- **Agile phù hợp với dự án như thế nào?**

- Agile phù hợp với các dự án đòi hỏi sự linh hoạt và có mức độ phức tạp hoặc không chắc chắn. Chẳng hạn, một sản phẩm hoặc dịch vụ chưa từng được nhóm xây dựng.
- Agile được sinh ra trong lĩnh vực phát triển phần mềm. Các giai đoạn trong mô hình Agile phù hợp với phát triển và kiểm thử phần mềm.



3. AGILE - Dự án linh hoạt

● Agile phù hợp với dự án như thế nào?

Để áp dụng thành công mô hình này cần một số điều kiện tiên quyết trong tổ chức:

- Thứ nhất, các thành viên phối hợp, giao tiếp hiệu quả trong nội bộ. Kỹ năng giao tiếp tốt giúp nhóm làm việc thấu hiểu khách hàng, hợp tác tốt với nhau đảm bảo chất lượng và tốc độ.
- Thứ hai, tính tự chủ của mỗi thành viên phải được đảm bảo để các nhóm tự quản lý có thể vận hành một cách chủ động, trơn tru thay vì chỉ tuân thủ theo chỉ dẫn cấp trên như trong các mô hình truyền thống.
- Thứ ba, các hoạt động được modul hóa thông qua những nhóm liên chức năng. Những nhóm này có khả năng làm việc với tốc độ và chất lượng cao, với khách hàng là trung tâm

4. Extreme Programming (XP)

- Là một trong những phương pháp phát triển phần mềm linh hoạt (Agile) phổ biến
- Nó tập trung vào việc cải thiện chất lượng phần mềm và tăng tính linh hoạt trong quá trình phát triển
- Đặt trọng tâm vào các nguyên tắc như phản hồi liên tục, sự tương tác giữa các thành viên trong nhóm và sự linh hoạt trong việc thay đổi yêu cầu.

4. Đặc trưng của XP

- **Phản hồi liên tục (Continuous Feedback):** Các phản hồi được thu thập thường xuyên từ khách hàng hoặc người dùng cuối, giúp điều chỉnh và cải thiện sản phẩm theo hướng phù hợp.
- **Kiểm thử liên tục (Continuous Testing):** Việc thực hiện kiểm thử trong suốt quá trình phát triển giúp đảm bảo chất lượng của sản phẩm và phát hiện các lỗi sớm.
- **Tương tác liên tục (Continuous Interaction):** Sự tương tác thường xuyên giữa các thành viên trong nhóm phát triển (như lập trình viên, kiểm thử viên và khách hàng) để đảm bảo sự hiểu biết đầy đủ về yêu cầu và tiến độ của dự án.
- **Lập trình cặp (Pair Programming):** Lập trình viên làm việc theo cặp, tạo ra một môi trường hợp tác và hỗ trợ, giúp cải thiện chất lượng mã nguồn và giảm thiểu số lượng lỗi.

4. Đặc trưng của XP

- **Tổ chức đơn giản (Simple Design):** Tập trung vào việc tạo ra một thiết kế đơn giản, dễ hiểu và dễ bảo trì.
- **Tạo ra mã sạch (Refactoring):** Liên tục cải thiện cấu trúc mã nguồn để giảm thiểu sự lặp lại và tăng tính linh hoạt của mã.
- **Lên lịch ngắn (Short Iterations):** Sử dụng các chu kỳ phát triển ngắn (ví dụ: 1-2 tuần) để tạo ra các phiên bản phần mềm nhỏ và linh hoạt.
- **Quản lý yêu cầu linh hoạt (Flexible Requirements):** Sự linh hoạt trong việc thay đổi yêu cầu của dự án để phản ánh sự thay đổi trong nhu cầu của khách hàng.

4. Extreme Programming (XP)

- Không thể lập trình cho đến khi biết mình đang làm gì.
- Cần khắc phục ở giai đoạn đầu tiên khi hệ thống đi vào sản xuất càng nhanh càng tốt. Tuy nhiên, mọi dự án đều phải bắt đầu từ điểm nào đó.
- Kết hợp phân tích tổng thể dưới dạng các vấn đề. Mỗi vấn đề phải theo định hướng kinh doanh, có thể kiểm tra và ước tính được.
- Một tháng là một khoảng thời gian dài để đưa ra những vấn đề cho một dự án.

Embracing Change with Extreme Programming, Beck, 1999

4. Extreme Programming (XP)

- Xác định thời điểm lập trình?
 - Khách hàng chọn bản phát hành tiếp theo tính năng có giá trị nhất (được gọi là các vấn đề trong XP) trong số tất cả các vấn đề có thể có, xác định bởi chi phí của các vấn đề và tốc độ của nhóm trong việc cài đặt chúng.
 - Khách hàng xác định vấn đề của lần lặp tiếp theo bằng cách chọn những vấn đề có giá trị nhất còn lại trong bản phát hành, được thông báo lại bằng chi phí của và tốc độ của nhóm.

Embracing Change with Extreme Programming, Beck, 1999

4. Extreme Programming (XP)

- Xác định thời điểm lập trình?
 - Các lập trình viên nhận trách nhiệm cho các nhiệm vụ chi tiết.
 - Xác định các trường hợp kiểm thử để chứng minh rằng nhiệm vụ đã hoàn thành.
 - Làm việc với đối tác để chạy các trường hợp kiểm thử, đồng thời cải tiến để duy trì một thiết kế đơn giản nhất có thể cho toàn bộ hệ thống.

Embracing Change with Extreme Programming, Beck, 1999

4. Extreme Programming (XP)

- Để XP thành công, cần có kiến thức chuyên môn.
 - Xây dựng, mô tả các vấn đề cần giải quyết của hệ thống đang được xây dựng.
 - Không mô tả một giải pháp, sử dụng ngôn ngữ kỹ thuật hoặc chứa các yêu cầu truyền thống.
 - Chuyển vấn đề thành mã.

4. Extreme Programming (XP)

- Biến thành mã thử nghiệm
- **Kiểm thử đơn vị là trọng tâm** của XP, trong đó hai điểm xoay quanh các chiến lược kiểm thử thông thường giúp kiểm tra hiệu quả hơn nhiều:
 - Các lập trình viên viết các test riêng và cần viết trước khi viết mã.
 - Thiết kế phát triển - Không được phá vỡ các thử nghiệm hiện có và cũng phải hỗ trợ phát triển gia tăng có thể mở rộng

4. Ưu điểm của XP

- Nếu được hoàn thành tốt, XP **cải thiện tính đồng đội**.
- **xây dựng năng lực** thực sự ở tất cả các thành viên trong nhóm.
- làm cho công việc thú vị.
- TDD (Test-driven development) hướng dẫn các nhà phát triển về cách viết mã chất lượng và cách cải thiện quan niệm về thiết kế; giúp cải thiện việc ước lượng.

4. Ưu điểm của XP

- Cung cấp **nhiều công cụ** cho quản lý , bao gồm khả năng dự đoán, tính linh hoạt của các nguồn lực, tính nhất quán và khả năng hiển thị về những gì thực sự đang diễn ra.
- Cung cấp cho khách hàng **khả năng xem xét** công ty có thể thực hiện công việc của mình hay không.
- **Không lãng phí thời gian** và không tạo ra nhiều tài liệu vô ích.

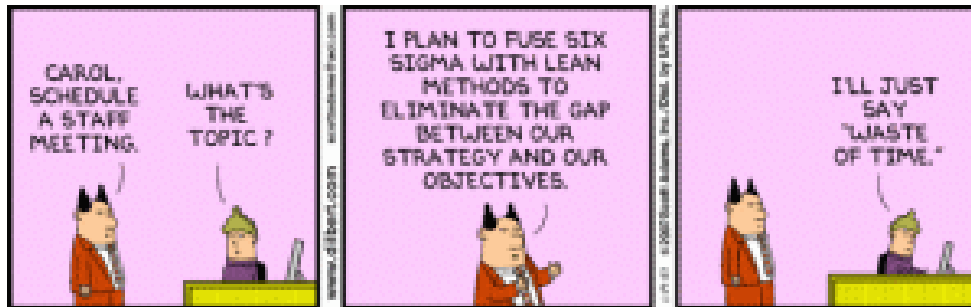
4. Nhược điểm của XP

- Thiết kế trở nên tiềm ẩn thay vì rõ ràng
- Dựa vào thiết kế là rủi ro
- Rất khó để viết các kiểm thử tốt
- Lặp lại quá thường xuyên có thể làm giảm chất lượng
- Để thực hiện tốt, cần phải làm thường xuyên

5. Lean – Phát triển phần mềm tinh gọn

J Paul Gibson: Agile Methods

October 2011



5. Lean Software Development

J Paul Gibson: Agile Methods

October 2011

- Mọi thứ sẽ dễ dàng hơn nếu khách hàng ngừng thay đổi suy nghĩ của họ.
- Hãy để khách hàng trì hoãn quyết định về những gì họ muốn càng lâu càng tốt và khi họ yêu cầu hãy để họ không có thời gian để thay đổi.
- Các thiết kế xuất hiện khi các nhà thiết kế phát triển sự hiểu biết về vấn đề, chứ không phải thu thập số lượng lớn các yêu cầu.
- Cung cấp hệ thống nhanh nhất có thể.

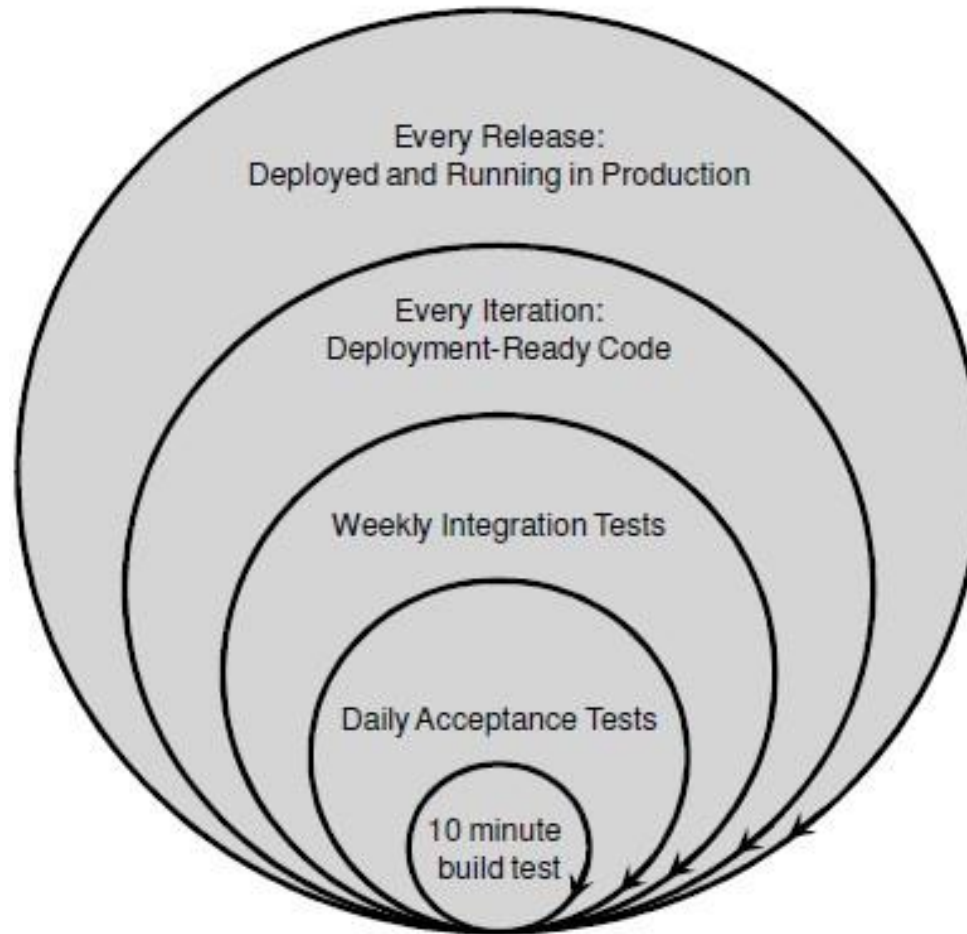
5. Lean Software Development

- **Eliminate waste.** Giảm thiểu sự lãng phí: Bất kỳ hoạt động nào không “tự trả giá” trong nỗ lực giảm thiểu ở những nơi khác trong hệ thống và cần được loại bỏ.
- **Amplify learning.** Nhấn mạnh vào việc học: Các nhà phát triển luôn cần học các phương pháp mới để tạo ra một hệ thống mạnh mẽ nhất.
- **Đưa ra quyết định muộn nhất có thể.** Tránh đưa ra quyết định sai lầm sớm và sau đó phải sửa chữa nó sau này.
- **Cung cấp nhanh nhất có thể.** Làm giảm đi việc thay đổi yêu cầu. Điều này có thể phải trả giá lớn nếu sự thay đổi đến muộn trong quá trình phát triển.

5. Lean Software Development

- **Để mọi người chịu trách nhiệm:** có động lực và gắn bó với nhau như một đội, cần phải chịu trách nhiệm về kết quả và được ủy quyền để biến nó thành hiện thực.
- **Duy trì tính toàn vẹn:** phải kiểm thử tích hợp, kiểm thử đơn vị và kiểm thử chung, đặc biệt là từ khách hàng.
- **Xem xét toàn bộ hệ thống:** đừng chia nhỏ hệ thống thành nhiều phần.

5. Tập trung vào kiểm thử

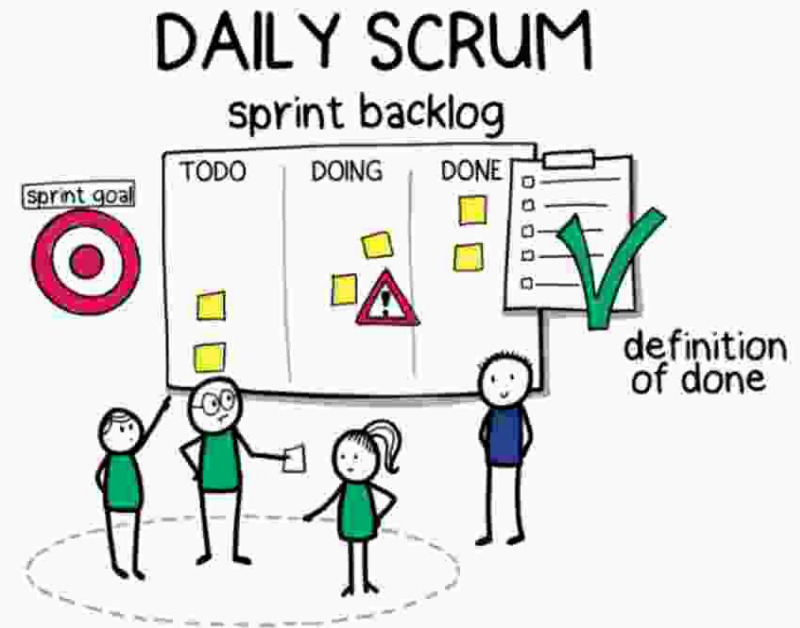


5. Lean Software Development

Ưu điểm	Nhược điểm
Tư duy toàn bộ hệ thống, mặc dù khó đối với hệ thống phức tạp, giúp đảm bảo tính nhất quán và toàn vẹn của hệ thống. Giảm thời gian tích hợp kể từ khi được phát triển.	Khó thực hiện việc hình dung cấu trúc của một hệ thống phức tạp là thông qua việc phân vùng hệ thống.
Nhóm làm việc thiết kế các quy trình của riêng mình, đưa ra các cam kết riêng, thu thập thông tin cần thiết để đạt được mục tiêu và tự lập chính sách để đạt được các mốc quan trọng của mình.	Lịch trình sẽ bị ảnh hưởng xấu bởi các quyết định muộn . Điều này có thể làm tổn hại đến sự phát triển song song và làm tăng thời gian thực hiện .

6. SCRUM - Quản lý dự án nước rút

- ‘Scrum’ là thuật ngữ chỉ một **nhóm người chơi với nhau** trong môn bóng bầu dục.
- Trong phát triển phần mềm, **công việc là đưa ra một bản phát hành.**
- Nhiệm vụ: **tăng tốc** việc phát hành sản phẩm



6. SCRUM - Quản lý dự án nước rút

- Quá trình phát triển được chia thành một loạt **các lần lặp được gọi là sprint**.
- Trước mỗi sprint, các thành viên trong nhóm **xác định các công việc còn tồn**.
- Khi kết thúc sprint, nhóm sẽ xem xét để **nêu rõ các kinh nghiệm và kiểm tra tiến độ**.



6. SCRUM - Quản lý dự án nước rút

- Scrum xuất phát là một quy trình phát triển phần mềm theo Agile. Vì thế, nó tuân thủ các nguyên tắc của Agile.
- Scrum là một khung quản lý dự án được áp dụng rất rộng rãi, bao gồm những dự án đơn giản với một nhóm phát triển nhỏ cho đến những dự án có yêu cầu rất phức tạp và kể cả những dự án đòi hỏi khung thời gian cố định
- Phương pháp Scrum là bộ khung làm việc giúp các công ty, tổ chức chia nhỏ công việc thành những phần nhỏ hơn, để quản lý dễ dàng hơn và được hoàn thành bởi một nhóm liên chức năng (cross-function) trong một khoảng thời gian quy định (còn gọi là Sprint trong 2-4 tuần).
- Nhóm Scrum thường sử dụng bảng để theo dõi công việc của từng thành viên trong nhóm (luồng chảy công việc – flow of work).

6. SCRUM - Quản lý dự án nước rút

● Ưu điểm của SCRUM :

- Các thành viên trong nhóm có một bức tranh rõ ràng hơn về nhiệm vụ của họ. Họ biết phải làm gì và bất kỳ câu hỏi nào có thể được giải quyết trong lần chạy nước rút tiếp theo của họ.
- Các dự án lớn được chia thành các phần nhỏ hơn để dễ quản lý và hoàn thành.
- Scrum là một khung có thể giúp bạn quản lý dự án của bạn hiệu quả hơn và sử dụng thời gian và ngân sách tốt hơn.
- Scrum là sự đảm bảo tính minh bạch của tất cả các giai đoạn của dự án.
- Một trong những nguyên tắc của Scrum là tập trung vào việc giảm thiểu lỗi. Nhờ cách tiếp cận này (ví dụ như chạy nhiều thử nghiệm), bạn có thể chắc chắn rằng dự án được duy trì ở mức chất lượng cao nhất.
- Scrum là một phương pháp rất linh hoạt. Nếu khách hàng muốn thực hiện bất kỳ thay đổi hoặc mở rộng sản phẩm với các chức năng mới, thường thì không có vấn đề gì với điều đó. Loại đàn hồi này được đảm bảo bằng nước rút.
- Các cuộc họp hàng ngày giúp xác định các mối đe dọa và vấn đề mới nổi có thể được giải quyết nhanh chóng.
- Scrum phù hợp với ngân sách, có thể thường xuyên kiểm soát tất cả các chi phí.

6. SCRUM - Quản lý dự án nước rút

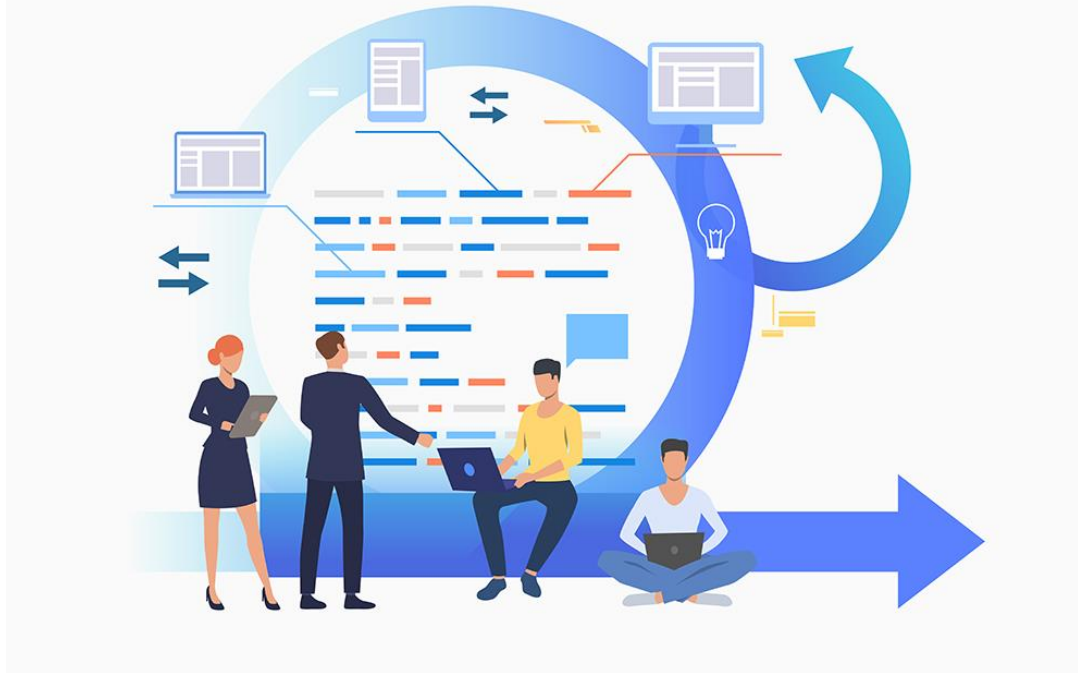
● **Nhược điểm của SCRUM :**

- Scrum phụ thuộc vào việc có những người lao động có năng lực, tận tâm và sẵn sàng làm việc như một phần của một nhóm. Thành công của dự án của bạn có thể gặp rủi ro nếu một thành viên trong nhóm không tham gia hoặc làm công việc của một người chậm hơn những người khác.
- Vai trò Scrum Master rất quan trọng. Nếu họ không thực hiện nhiệm vụ của mình một cách chuẩn chỉnh có thể dẫn đến sự chậm trễ trong dự án.
- Scrum thường chỉ lý tưởng cho các nhóm 3-9 người. Với các nhóm dự án lớn hơn có thể gặp vấn đề về hiệu quả quản lý.
- Các cuộc họp hàng ngày có thể gây khó chịu cho các thành viên trong nhóm.
- Sự ra đi bất ngờ của một thành viên trong nhóm có thể gây tổn hại cho tiến độ của toàn bộ dự án.
- Ngày giao sản phẩm và giới hạn thời gian chạy nước rút sẽ không áp dụng cho Scrum.

6. SCRUM - Quản lý dự án nước rút

● Scrum phù hợp với dự án như thế nào?

- Scrum thường được sử dụng trong việc phát triển những sản phẩm mà người dùng vẫn chưa xác định được mục tiêu cuối cùng.
- Người lãnh đạo của một đội sử dụng Scrum không giống như một huấn luyện viên đứng và chỉ đạo bên lề, và giống như đội trưởng trong trò chơi nhiều như mọi người khác.



6. SCRUM - Quản lý dự án nước rút

