



ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Nhóm chuyên môn Nhập môn Công nghệ phần mềm

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Các nguyên lý, nguyên tắc cơ bản của Agile



- 1. Cách tiếp cận của Agile**
- 2. Đặc trưng của Agile**
- 3. Các nguyên tắc của Agile**

MỤC TIÊU

Sau bài học này, người học có thể:

1. Hiểu cách tiếp cận khái niệm trong Agile trong phát triển phần mềm
2. Hiểu về các đặc trưng của Agile nói chung
3. Biết và áp dụng các nguyên tắc của Agile trong quy trình phát triển phần mềm

NỘI DUNG TIẾP THEO



1. Cách tiếp cận của Agile

1.1. Phát triển linh hoạt

1.2. Ví dụ một số phương pháp theo mô hình Agile

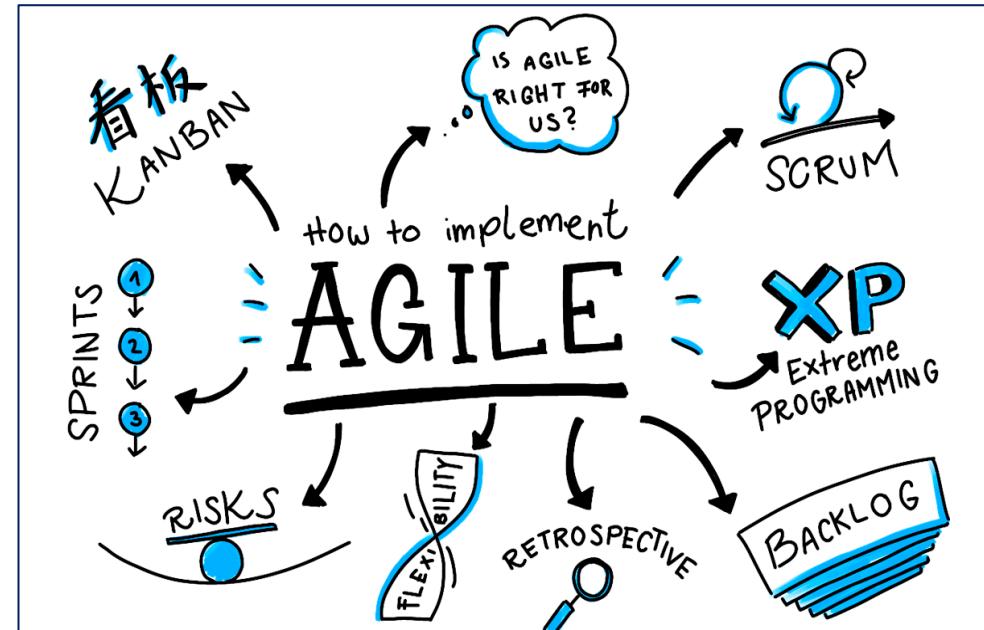
2. Đặc trưng của Agile

3. Các nguyên tắc của Agile

1. CÁCH TIẾP CẬN CỦA AGILE

1.1. Phát triển linh hoạt

- Agile không định nghĩa ra một phương pháp/cách thức cụ thể
- Có nhiều phương pháp khác nhau thỏa mãn và hướng theo các tiêu chí của Agile
- Thực tế: Nhiều công ty đã kết hợp các phương pháp lại với nhau



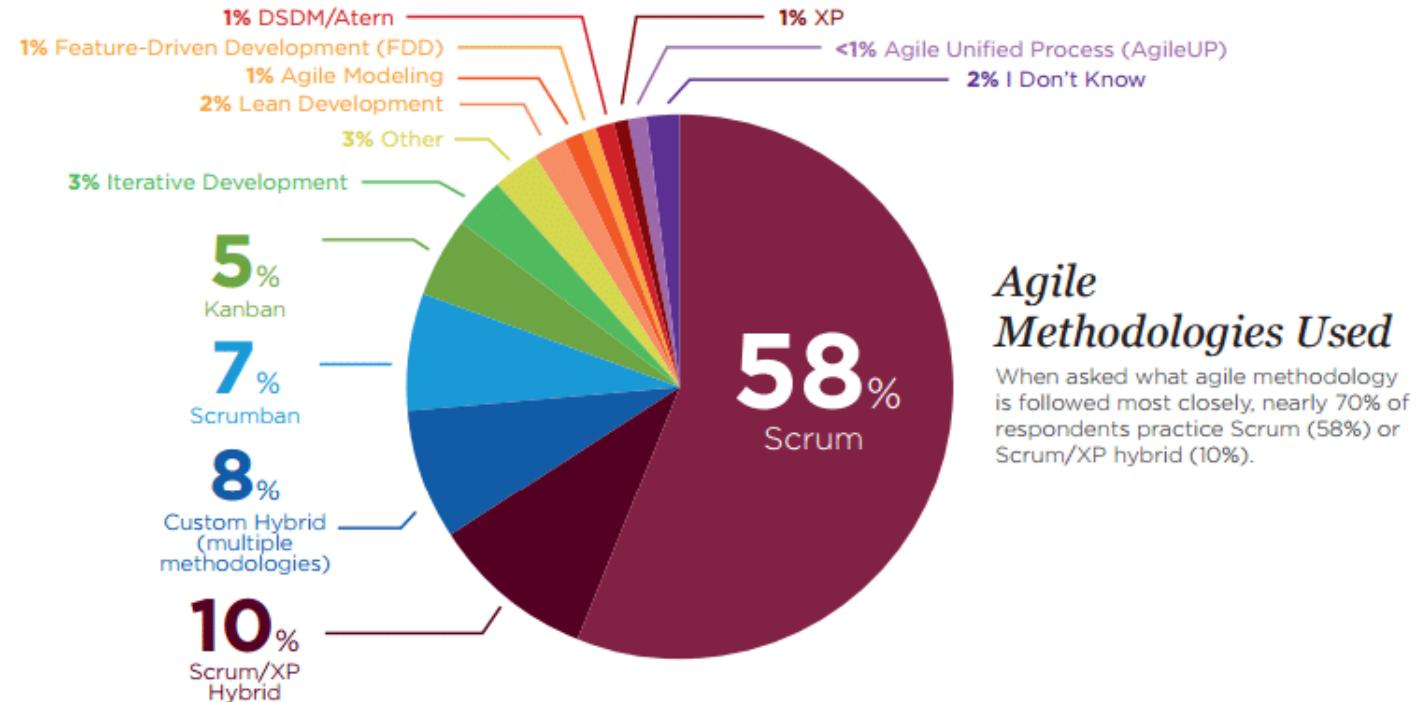
Hình 1.1: Agile – phương pháp phát triển phần mềm linh hoạt

Nguồn: [Elizabeth Venter](#), "Key learnings from implementing Agile across IT and Business functions

1. CÁCH TIẾP CẬN CỦA AGILE

1.2. Ví dụ một số phương pháp theo mô hình Agile

- Scrum
- Extreme programming (XP)
- Kanban
- Crystal methodology
- Feature-driven development (FDD)
-



Agile Methodologies Used

When asked what agile methodology is followed most closely, nearly 70% of respondents practice Scrum (58%) or Scrum/XP hybrid (10%).

Biểu đồ: Mức độ ưa chuộng sử dụng các phương pháp Agile trong doanh nghiệp

Nguồn: Budiman, Thomas & Suroso, Jarot. (2017). Optimizing IT Infrastructure by Virtualization Approach. IOP Conference Series: Materials Science and Engineering.

NỘI DUNG TIẾP THEO



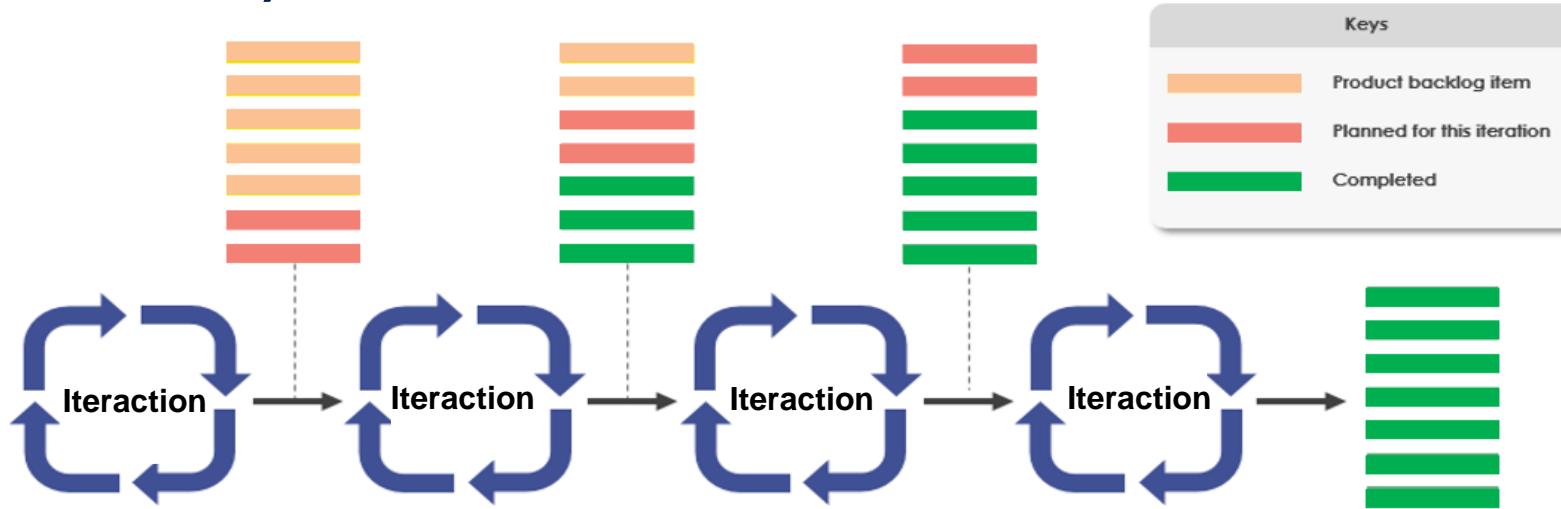
1. Cách tiếp cận của Agile

2. Đặc trưng của Agile

3. Các nguyên tắc của Agile

2. ĐẶC TRƯNG CỦA AGILE

▪ Tính lặp (Iteration)



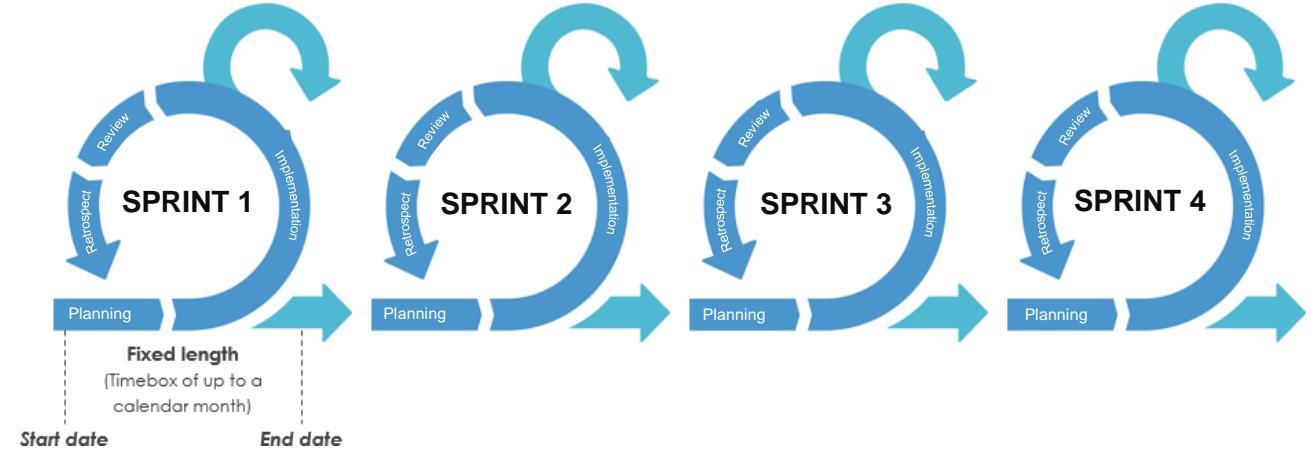
Sơ đồ: Các vòng lặp trong phương pháp Agile

- Dự án sẽ được thực hiện trong các phân đoạn lặp đi lặp lại (Iteration hoặc Sprint)
- Trong mỗi phân đoạn, thực hiện đầy đủ các công việc cần thiết như lập kế hoạch, phân tích yêu cầu, thiết kế, triển khai, kiểm thử.
- Khái niệm: Danh sách chức năng/công việc (**Product backlog item**) - danh sách các chức năng có thể được sắp xếp theo thứ tự ưu tiên mà một sản phẩm nên có

2. ĐẶC TRƯNG CỦA AGILE

▪ Tính tăng trưởng (Increments & Evolutionary)

- Mỗi bước lặp giống như phát triển một phần mềm hoàn chỉnh: xác định yêu cầu, phân tích thiết kế, viết mã, kiểm thử, viết tài liệu
- Kết quả sau mỗi giai đoạn phát triển đều được kiểm tra kĩ và sẵn sàng đưa vào sản phẩm

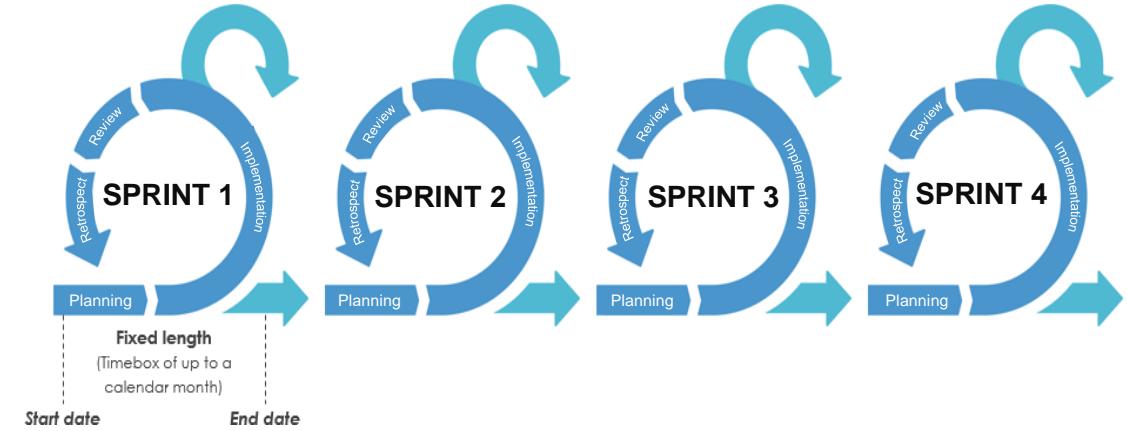


Mô hình: Các bước lặp (sprint) của Scrum

2. ĐẶC TRƯNG CỦA AGILE

▪ Tính tăng trưởng (Increments & Evolutionary)

- Dự án được chia thành nhiều phần/ giai đoạn phát triển nhỏ (sprint) nối tiếp nhau, sau mỗi giai đoạn, hệ thống được từng bước hoàn thiện
- Các phần/đoạn thời gian này thường kéo dài từ 2 đến 4 tuần
- Theo thời gian, các phần/đoạn nối tiếp nhau, sản phẩm của dự án được hoàn thiện dần tiến tới đáp ứng yêu cầu khách hàng
- Quá trình phát triển được thực hiện bởi nhóm nhỏ, thường từ 4 - 9 thành viên



Mô hình: Các bước lặp (sprint) của Scrum

2. ĐẶC TRƯNG CỦA AGILE

- **Tính thích ứng/thích nghi (Adaptive)**

- Kế hoạch sẽ liên tục được điều chỉnh
- Phù hợp theo các phân đoạn ngắn của dự án
- Kịp thời những yêu cầu thay đổi của khách hàng hay những tác động của các vấn đề khác

- **Nhóm tự tổ chức và liên chức năng**

- Nhóm tự tổ chức sẽ chịu trách nhiệm từng mảng công việc riêng biệt theo mỗi phân đoạn của dự án.
- Phù hợp với công việc được giao để có thể hoàn thành nhiệm vụ

2. ĐẶC TRƯNG CỦA AGILE

- **Quản lý tiến trình thực nghiệm (Empirical Process Control)**

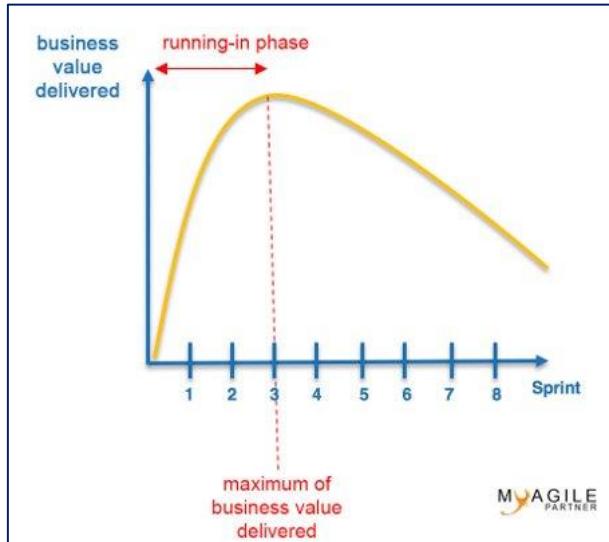
- Dựa vào dữ liệu thực tế để đưa ra các quyết định cho công việc
- Rút ngắn thời gian phản hồi và tăng tính linh hoạt

- **Giao tiếp trực tiếp (Face-to-face communication)**

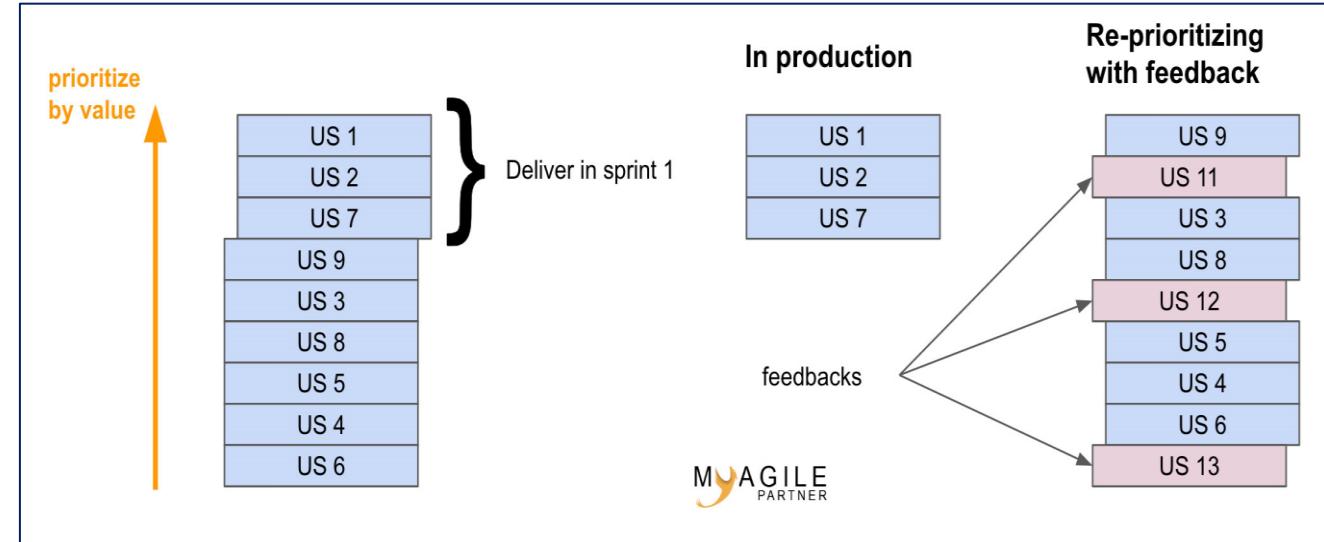
- Agile đánh giá cao việc trao đổi trực tiếp hơn là giao tiếp thông qua giấy tờ.
- Agile còn khuyến khích nhóm dự án trực tiếp nói chuyện với khách hàng để hiểu rõ điều họ đang cần.

2. ĐẶC TRƯNG CỦA AGILE

- Phát triển dựa trên giá trị (Value-based development)



Biểu đồ: Value Driven-Development (VDD)



Biểu đồ: Value Driven-Development (VDD)

- Thường xuyên trao đổi với khách hàng
 - Nắm được những yêu cầu có mức độ ưu tiên cao
 - Đưa ra những điều chỉnh phù hợp nhằm đem lại giá trị sớm nhất

Nguồn: Judicaël Paquet, Value Driven-Development (VDD), myagilepartner

NỘI DUNG TIẾP THEO



1. Cách tiếp cận của Agile
2. Đặc trưng của Agile
- 3. Các nguyên tắc của Agile**

3. CÁC NGUYÊN TẮC CỦA AGILE



▪ 12 nguyên tắc của mô hình Agile

- Sự hài lòng của khách hàng cần đặt lên hàng đầu
- Luôn thay đổi khi cần thiết trong quá trình phát triển
- Thường xuyên ra mắt phần mềm làm việc, ưu tiên trong khung thời gian ngắn
- Luôn hợp tác với những bên liên quan cùng với bộ phận phát triển kinh doanh.
- Những cá nhân có động lực được làm việc trong môi trường có sự hỗ trợ tốt, giúp mang lại hiệu quả công việc
- Trao đổi trực tiếp.

3. CÁC NGUYÊN TẮC CỦA AGILE



▪ 12 nguyên tắc của mô hình Agile (tiếp)

- Phần mềm hoạt động (working software) là thước đo quan trọng nhất để đo sự tiến bộ.
- Thúc đẩy sự phát triển một cách bền vững, có tính ổn định liên tục.
- Nâng cao sự linh hoạt của chi tiết kỹ thuật và thiết kế.
- Đề cao tính đơn giản.
- Kiến trúc, yêu cầu, thiết kế tốt nhất được tạo ra từ các nhóm tự tổ chức (soft-organizaing teams).
- Điều chỉnh hành vi phù hợp.

TỔNG KẾT VÀ GỢI MỞ



1. Bài học đã cung cấp cho người học một số nội dung quan trọng của Agile, bao gồm **cách tiếp cận, đặc trưng** cũng như **các nguyên tắc** trong Agile
2. Tiếp sau bài này, **người học có thể tự tìm hiểu thêm** về ứng dụng của Agile cũng như các ưu, nhược điểm của phương pháp này.

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Các nguyên lý, nguyên tắc cơ bản của Agile

Biên soạn:

TS. Nguyễn Nhất Hải

Trình bày:

TS. Nguyễn Nhất Hải





NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Bài học tiếp theo:

Ưu, nhược điểm của phương pháp Agile

Tài liệu tham khảo:

- [1] R. Pressman, Software Engineering: A Practitioner's Approach. 8th Ed., McGraw-Hill, 2016.
- [2] I. Sommerville, Software Engineering. 10th Ed., AddisonWesley, 2017.
- [3] Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Ed., Springer.
- [4] Shari Lawrence Pleeger, Joanne M. Atlee, Software Engineering theory and practice. 4th Ed., Pearson, 2009

Tư liệu:

- [1] PMBOK® Guide – Seventh Edition. PMI, 2021.