* Chương 5
* Phương pháp quy trình mới và mới nổi
* Mùa thu năm 2011

Mục tiêu

* Hiểu không
  + Hạn chế của phương pháp quy trình truyền thống
  + Khả năng ứng dụng của các quy trình nhanh
  + Nguyên lý cơ bản của quy trình nhanh
* Làm quen với một số quy trình nhanh

Các quy trình nhanh là gì

* Họ phương pháp phát triển phần mềm
* Bản phát hành ngắn và lặp lại
  + Chia thành nhiều phần nhỏ
  + Phát hành cho khách hàng thường xuyên nhất có thể
* Thiết kế gia tăng
  + Đừng cố hoàn thành thiết kế trước
  + Trì hoãn các quyết định thiết kế càng lâu càng tốt
* Sự tham gia của người sử dụng
  + Nhận phản hồi của người dùng thường xuyên nhất có thể
* Tài liệu tối thiểu
  + Chỉ cần một lượng tài liệu cần thiết
* Truyền thông không chính thức
  + Mọi người giao tiếp tốt hơn không chính thức
* Thay đổi
  + Mọi thứ sẽ thay đổi, lên kế hoạch cho điều đó
  + Gia đình của chúng tôi
  + Phiên bản phát hành và phiên bản mới
  + Chia thành
  + Phát hành cho
  + Thiết Kiến gia Gia
  + Trong phần còn lại của chúng tôi
  + Chiêu thì tối giản, đẹp mắt
  + Gia tham gia của người đẹp
  + Tinh thần của bạn
  + Tài liệu
  + Chỉ số tài liệu
  + Giao tiếp không chính thống
  + Một trong hai
  + Thay đổi
  + Phần cứng của họ thay đổi, phần mềm

Tuyên ngôn Agile

* Chúng tôi đang khám phá những cách tốt hơn để phát triển phần mềm bằng cách làm điều đó và giúp người khác làm điều đó.
* Đó là, trong khi có giá trị trong các mục bên phải, chúng tôi đánh giá các mục bên trái nhiều hơn.
* [http://agilemanifesto.org](https://translate.google.com/translate?hl=vi&prev=_t&sl=auto&tl=vi&u=http://agilemanifesto.org/)
* Phần cứng và phần cứng, phần cứng và hình chữ
* Mạnh là, trong khi có giá trị trong khi có
* 1. Cá phiên và ngược dòng - - tài khoản của họ và dữ liệu. - Sau sau khi Cáp
* 1. Cá nhân và tương tác - - qua các quy trình và công cụ 2 . Phần mềm làm việc - - trên tài liệu toàn diện 3 . Hợp tác khách hàng - - đàm phán hợp đồng 4 . Đáp ứng với thay đổi - - theo kế hoạch

Các vấn đề với quy trình truyền thống

* Thời gian phát triển dài
* Không có khả năng đối phó với những thay đổi trong yêu cầu
* Giả định các yêu cầu được hiểu khi bắt đầu dự án
* Dựa vào nỗ lực phát triển anh hùng
* Phương pháp phức tạp
* Lãng phí / trùng lặp nỗ lực

Một số phương pháp Agile

* Không có quy trình sẽ làm việc cho tất cả các dự án
* Chọn quy trình phù hợp nhất với dự án
* Bốn trong số nhiều phương pháp nhanh
  + Lập trình cực đoan (XP)
  + Crystal Clear / Orange
  + Quy trình hợp nhất (RUP)
    - Khung
  + Khung giải pháp Microsoft
  + Không có quy trình
  + Bạn có thể chọn
  + Số lượng lớn
  + Lập kế hoạch đo lường (XP)
  + Crystal Clear / Orange
  + -Quy tích hợp tổng hợp lý
  + Khung
  + Khung giải pháp Microsoft

XP: Giá trị cốt lõi

* *Giao tiếp* (giữa nhóm và với khách hàng)
* *Đơn giản* (trong thiết kế và mã)
* *Phản hồi* (ở nhiều cấp độ)
* *Can đảm (đưa ra và thực hiện quyết định khó khăn)*

XP: Nguyên tắc cơ bản

* Phản hồi nhanh
  + Sử dụng lập trình cặp, kiểm tra đơn vị, tích hợp, và các bản phát hành và tương tác ngắn
* Sự đơn giản
  + Hãy thử cách tiếp cận đơn giản nhất có thể
* Gia tăng
  + Hãy thử những thay đổi nhỏ cộng lại
  + Sử dụng tái cấu trúc mã - sửa đổi nhỏ để cải thiện mã
* Nắm lấy thay đổi
  + Bảo tồn các lựa chọn cho tương lai
  + Trì hoãn các quyết định cam kết một đến một con đường càng lâu càng tốt
* Công việc có chất lượng
  + Tạo ra sản phẩm tốt nhất có thể

XP: Các nguyên tắc khác

* Học tập liên tục
* Đầu tư ban đầu nhỏ
* Chơi để chiến thắng
* Thí nghiệm bê tông
* Giao tiếp cởi mở, trung thực
* Làm việc với bản năng của mọi người
* Chấp nhận trách nhiệm
* Thích ứng địa phương
* Sự làm sáng tỏ trong ngành du lịch
* Đo lường trung thực

XP: 12 thực hành

* Lập kế hoạch
  + Xác định các tính năng được bao gồm trong phiên bản tiếp theo
  + Ưu tiên kinh doanh và ước tính kỹ thuật
* Bản phát hành ngắn
  + Nhận hệ thống làm việc nhanh chóng
  + Phát hành phiên bản mới trong chu kỳ ngắn (2 đến 4 tuần)
  + Kế hoạch chi tiết mới về phản hồi của khách hàng
* Ẩn dụ
  + Sử dụng một phép ẩn dụ thay vì kiến ​​trúc
* Thiết kế đơn giản
  + Loại bỏ sự phức tạp không cần thiết càng sớm càng tốt
  + Thiết kế có thể được thay đổi trong các phiên bản sau
* Hướng phát triển thử nghiệm
  + Liên tục và tự động
  + Viết bài kiểm tra trước khi viết mã
* Cải tiến thiết kế (Tái cấu trúc)
  + Xóa trùng lặp, cải thiện giao tiếp và đơn giản hóa hoặc thêm tính linh hoạt cần thiết

XP: 12 thực hành

* Lập trình cặp
  + Hai lập trình viên làm việc cùng một máy
* Sở hữu tập thể
  + Bất cứ ai cũng có thể thay đổi bất kỳ mã nào bất cứ lúc nào
* Hội nhập liên tục
  + Tích hợp và kiểm tra mỗi khi hoàn thành một nhiệm vụ (nhiều lần trong ngày)
* Bước đi bền vững
  + 40 giờ một tuần hợp lý
  + Không bao giờ làm thêm hai tuần liên tiếp
* Khách hàng tại chỗ
  + Bao gồm khách hàng thực sự trong nhóm
* Tiêu chuẩn mã hóa
  + Mọi người cần sử dụng các quy tắc giống nhau

XP: Lập kế hoạch

* Lập kế hoạch chỉ lặp lại ngay lập tức chi tiết
* Bốn biến để điều chỉnh trong một dự án
  + Phạm vi
    - Những gì cần phải được thực hiện để hệ thống có ích
  + Giá cả
  + Phẩm chất
  + Thời gian
* Phạm vi dễ thay đổi hơn (bỏ tính năng)
* Sử dụng các câu chuyện (thể hiện một nhiệm vụ, tương tự như các trường hợp sử dụng)

XP: Lập kế hoạch (2)

* Khách hàng quyết định
  + Phạm vi: cần thiết cho hệ thống hữu ích
  + Ưu tiên: xác định và ưu tiên các đặc điểm
  + Phạm vi phát hành: những gì được bao gồm trong mỗi bản phát hành
  + Ngày phát hành
* Bên kỹ thuật quyết định
  + Ước tính: mỗi tính năng sẽ mất bao lâu
  + Hậu quả: công nghệ tốt nhất và ngôn ngữ lập trình
  + Quy trình: các hoạt động công việc được thực hiện như thế nào; tổ chức nhóm
* Ai đưa ra quyết định?

XP: Trò chơi lập kế hoạch

* Mục tiêu: Tối đa hóa giá trị của phần mềm được sản xuất
* Chiến lược: Đầu tư càng ít càng tốt để có được hầu hết các chức năng càng nhanh càng tốt
* Ba giai đoạn:
  + Thăm dò - phân định phạm vi có thể
  + Cam kết - chọn phạm vi phát hành, ngày
    - Kinh doanh sắp xếp các câu chuyện (tính năng) thành ba loại: cần thiết, quan trọng và tốt đẹp để có
    - Phát triển sắp xếp các câu chuyện theo rủi ro: có thể ước tính chính xác, có thể ước tính hợp lý tốt, những câu chuyện không thể ước tính
    - Doanh nghiệp chọn phạm vi, những thẻ câu chuyện có thể đáp ứng lịch trình hoặc lịch biểu được điều chỉnh để đáp ứng phạm vi
  + Chỉ đạo - điều chỉnh kế hoạch
    - Thêm, xóa hoặc điều chỉnh câu chuyện

Trò chơi lập kế hoạch XP - Thăm dò

* Mục tiêu: Tìm hiểu / quyết định những gì hệ thống có thể làm
* Di chuyển
  + Viết một câu chuyện (khách hàng)
  + Ước tính một câu chuyện (nhà phát triển)
  + Chia một câu chuyện

Trò chơi lập kế hoạch XP - cam kết

* Doanh nghiệp chọn phạm vi và ngày phát hành tiếp theo
* Phát triển cam kết với nó
* Di chuyển
  + Sắp xếp các câu chuyện (cần thiết, quan trọng, tốt đẹp)
  + Sắp xếp theo rủi ro (ước tính chính xác, đoán, không có đầu mối)
  + Phát triển thiết lập vận tốc (lý tưởng so với thời gian lịch)
  + Doanh nghiệp chọn phạm vi (câu chuyện / nhiệm vụ)

Trò chơi lập kế hoạch XP - Chỉ đạo

* Cập nhật và điều chỉnh kế hoạch
* Di chuyển
  + Khách hàng có thể chọn câu chuyện để lặp lại
  + Phục hồi
    - Nhà phát triển yêu cầu khách hàng sắp xếp lại
  + Thêm / Thay thế câu chuyện
  + Ước tính lại

Pha lê

* Một phương pháp không thể phù hợp với tất cả các dự án
* Điều gì / làm thế nào để thích ứng với dự án
* Phân loại dự án theo
  + Kích thước (số lượng nhà phát triển)
  + Tính quan trọng: Mất mát mà trục trặc sẽ gây ra
    - Đời sống
    - Tiền thiết yếu
    - Tiền tùy ý
    - Sự thoải mái của khách hàng
  + Ưu tiên (áp lực thời gian)

Ba phương pháp xác định

* Màu càng đậm, phương pháp càng nặng
  + Tinh thể rõ ràng
    - Dự án không quan trọng
    - Các đội từ sáu đến tám
  + Pha lê cam
    - Quan trọng nhưng không quan trọng cuộc sống
    - Đội lên tới 40 người
  + Crystal cam web
* Cần nhiều hơn nữa ba
  + Quan trọng cuộc sống
  + Dự án quy mô lớn
  + V.v.

Điều chỉnh phương pháp

* Phương pháp lớn hơn cho các đội lớn hơn
* Phương pháp nặng hơn cho các dự án quan trọng hơn
* Ưu tiên cho giao tiếp tương tác trên tài liệu chính thức
* Hiểu rằng mọi người khác nhau. Các quy trình kỷ luật cao khó áp dụng hơn
* Giả sử mọi người muốn trở thành 'công dân tốt'

Pha lê - Thuộc tính

* Giao hàng thường xuyên
  + Ít nhất vài tháng một lần
  + Các loại giao hàng: tất cả người dùng, người dùng hạn chế, chỉ demo
* Cải thiện phản xạ
  + Luôn suy nghĩ những gì có thể được cải thiện
* Giao tiếp chặt chẽ
  + Các thành viên trong nhóm gần nhau
* An toàn cá nhân
  + Không trả thù vì đã lên tiếng
* Tiêu điểm
  + Giảm thiểu gián đoạn
* Dễ dàng truy cập đến người dùng chuyên gia
  + Nhận chuyên gia tư vấn khi cần
* Môi trường kỹ thuật tốt
  + Kiểm tra tự động, quản lý cấu hình và tích hợp thường xuyên

|  |  |  |
| --- | --- | --- |
|  | * Thông thoáng | * trái cam |
| * Đội | * Một đội, cùng phòng | * Các nhóm khác nhau để lập kế hoạch hệ thống, giám sát dự án, kiến ​​trúc, công nghệ, chức năng, cơ sở hạ tầng và thử nghiệm bên ngoài |
| * Vai trò / Người riêng biệt | * Ít nhất 4 người, đóng vai trò Nhà tài trợ, Nhà thiết kế cao cấp, lập trình viên, người dùng. * Các vai trò khác có thể được thực hiện bởi cùng một người, bao gồm Quản lý dự án, Chuyên gia kinh doanh, Người thu thập yêu cầu | * 14 vai trò khác nhau, được chơi bởi những người khác nhau, bao gồm (ngoài vai trò của Crystal Clear), Giám đốc dự án, Nhà tài trợ, Chuyên gia kinh doanh, Kiến trúc sư, Người thiết kế, Người thử nghiệm và Nhà thiết kế UI. |
| * Công việc sản xuất | * 9 mục, bao gồm lịch trình, trường hợp sử dụng, bản phác thảo thiết kế, trường hợp thử nghiệm và hướng dẫn sử dụng. | * 13 mục, bao gồm các mục trong Crystal Clear cộng với các tài liệu yêu cầu, báo cáo trạng thái, tài liệu thiết kế giao diện người dùng và thông số kỹ thuật giữa các nhóm. * Các sản phẩm làm việc được phát triển cho đến khi chúng dễ hiểu, chính xác và đủ ổn định để đánh giá ngang hàng. |
| * Thời lượng phát hành tối đa | * 2 tháng | * 2-4 tháng, hai lần xem người dùng mỗi lần phát hành |

* Hình 5.2: So sánh Crystal Clear và Crystal Orange

Pha lê - Đặc điểm chung

* Tiến độ được theo dõi bởi việc giao hàng, không phải bằng tài liệu
* Kiểm tra hồi quy tự động
* Sự tham gia của người dùng trực tiếp
* Hai lượt xem cho mỗi người dùng
* Hội thảo điều chỉnh phương pháp
* Tiêu chuẩn chính sách bắt buộc
* Kiểu mã hóa, vv để lại các tiêu chuẩn địa phương
* Kỹ thuật cho các vai trò cá nhân được để lại cho cá nhân
* Chúng tương tự như XP

Quy trình hợp nhất là Agile

* UP thường được coi là nặng
  + Tất cả các yêu cầu khi bắt đầu
  + Kiến trúc, thiết kế trả trước
  + Quy trình hợp nhất (RUP) hợp lý bắt buộc nhiều tài liệu / tạo tác
* Nhưng
  + lặp đi lặp lại và gia tăng
  + UP không thực sự yêu cầu tất cả các sản phẩm, RUP không
* Hạn chế sản phẩm, loại bỏ vai trò, thêm các lần lặp

Khung giải pháp Microsoft

* Khung, không phải phương pháp (tăng một cấp)
* Có thể được thực hiện nhanh nhẹn
* Các thành phần
  + Nguyên tắc nền tảng
  + Mô hình
  + Kỷ luật
  + Ý chính
  + Thực hành đã được chứng minh
  + khuyến nghị

MSF: Nguyên tắc nền tảng

* Thúc đẩy truyền thông mở
* Làm việc hướng tới một tầm nhìn chung
* Trao quyền cho các thành viên trong nhóm
* Thiết lập trách nhiệm rõ ràng và trách nhiệm chung
* Tập trung vào giá trị kinh doanh
* Luôn nhanh nhẹn, mong đợi sự thay đổi
* Đầu tư vào chất lượng
* Học hỏi từ tất cả các kinh nghiệm

Mô hình nhóm MSF

|  |  |
| --- | --- |
| * Vai trò | * Mục tiêu chất lượng chính |
| * Quản lý chương trình | * Giao hàng trong phạm vi dự án |
| * Phát triển | * Giao hàng tận nơi |
| * Kiểm tra | * Đảm bảo mỗi bản phát hành giải quyết tất cả các vấn đề |
| * Quản lý phát hành | * Triển khai suôn sẻ và quản lý liên tục |
| * Kinh nghiệm người dùng | * Tăng cường hiệu suất người dùng |
| * Quản lý sản phẩm | * Khách hàng hài lòng |

Mô hình quy trình MSF

|  |  |  |
| --- | --- | --- |
| * Giai đoạn | * Cột mốc (có thể giao hàng) | * Tư duy |
| * Hình dung | * Tầm nhìn / phạm vi được phê duyệt | * Thăm dò |
| * Lập kế hoạch | * Kế hoạch dự án đã được phê duyệt | * Điều tra |
| * Phát triển | * Phạm vi hoàn thành (tất cả các tính năng được thực hiện) | * Thiết kế sáng tạo) |
| * Ổn định | * Sẵn sàng phát hành đã được phê duyệt (hoàn thành chức năng sửa lỗi) | * Single-Minded (tìm và sửa lỗi) |
| * Triển khai | * Triển khai hoàn tất (cho phiên bản này) | * Kỷ luật |

MSF - Kỷ luật quản lý

* Quản lý dự án
* Quản lý rủi ro
* Quản lý sẵn sàng

Phát triển nguồn mở

* Mở nguồn, cho phép người dùng sửa đổi
* Nhiều nhà phát triển trả lại mã, cải thiện chương trình
* Câu chuyện thành công: Linux, Apache, MySQL
* Không thực sự nhanh nhẹn

Mã nguồn mở so với Agile

* Điểm tương đồng
  + Bản phát hành nhỏ
  + Truyền thông không chính thức
  + Sẵn có của khách hàng
  + Hội nhập liên tục
  + Tầm nhìn chung
* Sự khác biệt
  + Các đội lớn hơn
  + Đội phân phối
  + Thu nhỏ

Agile vs truyền thống

|  |  |  |
| --- | --- | --- |
|  | * Nhanh nhẹn | * Truyền thống / Nặng |
| * Yêu cầu | * Giả định thay đổi * Yêu cầu không chính thức * Tương tác người dùng liên tục | * Giả sử không có thay đổi * Tài liệu yêu cầu đầy đủ, chi tiết, chính thức |
| * Thiết kế | * Không chính thức * Lặp đi lặp lại | * Chính thức * Trả trước |
| * Sự tham gia của người sử dụng | * Quan trọng * Thường xuyên | * Bắt đầu (Yêu cầu) * Kết thúc (Kiểm tra chấp nhận) |
| * Tài liệu | * Tối thiểu, chỉ khi cần thiết * Mã nguồn. | * tài liệu nặng, chính thức |
| * Giao tiếp | * Không chính thức * Xuyên suốt dự án. | * Các tài liệu * Ghi nhớ chính thức và các cuộc họp |
| * Phức tạp | * Thấp | * Cao. |
| * Trên không | * Thấp | * Cao |

Quá trình so với dự án

|  |  |  |
| --- | --- | --- |
|  | * *Nhanh nhẹn (XP)* | * *Truyền thống (RUP)* |
| * Phạm vi / kích thước | * Nhỏ bé. * Một đội, 3-10 người. | * Dự án lớn. * Có thể thu nhỏ lại |
| * Quan trọng | * Tương đối thấp. * Không dành cho các hệ thống quan trọng trong cuộc sống | * Cao hơn |
| * Những người | * Đội chơi * 'Công dân tốt' * Nhà phát triển giỏi | * Hầu như bất cứ ai * Vai trò khác nhau |
| * Văn hóa công ty | * Công ty nhỏ có cùng vị trí * Văn hóa thư giãn. | * Công ty lớn hơn * Địa điểm từ xa * Văn hóa chính thức. |
| * Ổn định | * Không cần thiết * Bản sao với những thay đổi trong yêu cầu hoặc môi trường. | * Giả định môi trường ổn định và yêu cầu * Ít phù hợp để đối phó với những thay đổi. |

Nhanh nhẹn so với truyền thống

* Ưu điểm
  + Đơn giản hơn
  + Chi phí thấp
  + Thỏa thuận với những thay đổi
  + Kết quả nhanh
  + Hệ thống có thể sử dụng
* Rủi ro, bất lợi
  + Không thể mở rộng
  + Dựa vào tinh thần đồng đội
  + Dựa vào quyền tiếp cận khách hàng
  + Xung đột văn hóa