Lập trình cực độ (eXtreme Programming viết tắt là XP) là một trong những nhóm các phương pháp phát triển phần mềm một cách linh hoạt. XP sử dụng các nhóm làm việc kết hợp gồm những người lập trình, khách hàng và các nhà quản trị để phát triển phần mềm có chất lượng cao trong thời gian nhanh chóng. Một chương trình chạy được là thước đo đầu tiên của tiến trình theo XP. XP có thể phát triển và tồn tại được là do sự hiểu biết ngày một tiến bộ về các vấn đề đang giải quyết và cũng là vì các công cụ sẵn có cho phép ta thay đổi được cái giá của sự thay đổi (cost-of-change) (đang là dạng hàm mũ trước đây). XP giữ cho cái giá phải trả này ở mức thấp do vậy sẽ thúc đẩy môi trường sản xuất phần mềm.

Ưu Điểm

Như ta đã biết hầu hết các phương pháp đều xem xét việc phát triển phần mềm như là một quy trình gia công với tiến trình viết phần mềm đi theo một con đường

Nhu cầu (thị trường) – Phân tích – Thiết kế – Viết mã nguồn – Thử nghiệm – Bảo trì

Cách tiếp cận này có một sự thừa nhận quan trọng đó là ta đã biết được sản phẩm cuối cùng trước khi tiến trình bắt đầu. Nhưng hầu hết các dự án phần mềm hiện đại không thể thoả mãn cái sự thừa nhận này. Khách hàng sẽ đưa ra một cách đầy đủ những cái gì mới và người sản xuất cần những thông tin phản hồi một cách liên tục để đánh giá lại các lựa chọn của họ. Người lập trình cần phải có một phương án để luôn sẵn sàng đón nhận những thay đổi trong Nhu cầu để họ có thể đối phó được với các thông tin phản hồi. Nếu bạn làm việc trong một môi trường mà ở đó các Nhu cầu được chờ để thay đổi và các khách hàng sẽ được lợi từ việc bàn giao phần mềm sớm và thường xuyên thì chắc chắn nên xem xét XP. Các nhóm làm theo XP sẽ thường xuyên nhận ra rằng họ đang bàn giao các sản phẩm phần mềm chất lượng cao với số lượng rất lớn và nhanh hơn trước đây rất nhiều.

Phương thức tiến hành XP

XP là một phương pháp có khả năng thích nghi, thích ứng. Điều đó có nghĩa là sẽ không có hai dự án XP nào giống nhau cả. XP cung cấp một tập hợp các thực hành và được sử dụng như là điểm khởi đầu, và sau đó được làm cho thích ứng để phù hợp với các ràng buộc của từng dự án riêng. Sau đây là một tập hợp các nguyên tắc được mong đợi:

Liên lạc: một nhóm XP lớn mạnh dựa trên các kiến thức, sự hiểu biết bài toán và hiểu biết phần mềm được chia sẻ. Các phương pháp giải quyết vấn đề được trao đổi trực tiếp. Những thứ cản trở đến công việc đều được loại bỏ.

Đơn giản hoá: công việc cần giảm tối thiểu độ phức tạp để đạt được hiệu quả cao nhất.

Thông tin phản hồi thường các đội làm dự án và khách hàng của họ không nhận ra những vấn đề rắc rối cho tới khi sắp bàn giao sản phẩm.Nhưng các nhóm XP thường xuyên lấy yêu cầu trở trong quá trình làm việc, thử, bàn giao sản phẩm … Khi đó sẽ quản trị được các vấn đề phát sinh.

Thế mạnh: các đội làm phần mềm thành công cần phải kiểm soát được ngay cả khi xuất hiện các lỗi. XP đưa ra 12 phương án thực hành, và điểm mạnh của XP chính là đã kết hợp được các phương án này lại. Mỗi một phương án tuy đơn giản nhưng rất cần thiết phải nắm vững, sẽ góp phần làm giảm bớt đáng kể cái giá của sự thay đổi.

Các phương án tiến hành

Tiêu chuẩn hoá mã nguồn

Đây là một loạt các quy ước về mã hoá để các thành viên của dự án theo đó làm. Khi đó mọi người có thể xem xét lẫn nhau và có thể bàn giao được cho nhau.

Thiết kế đơn giản

Để giảm giá phải trả cho sự thay đổi đồng nghĩa với việc làm cho hệ thống càng đơn giản càng tốt. Điều đó có nghĩa là không nên bỏ quá nhiều thời gian và công sức vào những việc mà sau này có thể cần hoặc cũng có thể không. Các đề án XP thường được đơn giản một cách tối đa, đảm bảo rằng sau này nếu có cần thay đổi thì chi phí cũng rất nhỏ.

Dạng thức Thiết kế và Nguyên lý Thiết kế

Mọi thành viên trong nhóm đều phải học và sử dụng thành thạo các Nguyên lý và Dạng thức Thiết kế. Thứ nhất, nó giúp cho cả nhóm làm việc với nhau một cách ăn ý (liên lạc tốt). Sau đó là giúp cho việc viết mã của từng thành viên được tốt và nhanh do tái sử dụng được kinh nghiệm từ người đi trước, điều này rất quan trọng, vì trong XP không có thiết kế chi tiết, từng đoạn mã/từng module phải do từng thành viên của nhóm thể hiện, vì vậy nếu áp dụng được thì sẽ giảm thiểu được quá trình điều chỉnh/tái chế.

--Trinhhoa (thảo luận) 04:18, ngày 21 tháng 12 năm 2011 (UTC)===Thử nghiệm (thử thiết kế đầu tiên)=== Các lập trình viên sẽ viết các đơn vị thử nghiệm (unit test) (trường hợp thử nghiệm) trước khi viết mã (thiết kế thử nghiệm đầu tiên). Tất cả các đơn vị thử nghiệm (trường hợp thử nghiệm) đều được thực hiện thường xuyên trong quá trình phát triển sản phẩm trong từng module mã của từng lập trình viên. Các lập trình viên có thể thay đổi một cách linh hoạt vì quy trình thử nghiệm có thể mắc lỗi hay sai so với thiết kế ban đầu (điều chỉnh trường hợp thử nghiệm).

Lập trình cặp đôi

Tất cả quá trình phát triển phần mềm đều được thực hiện bởi 2 người trên cùng một máy tính. Điều này có nghĩa là 2 người cùng giải quyết 1 công việc, người này lập trình thì người kia làm giám sát và ngược lại. Khi đó việc trao đổi kinh nghiệm và giải quyết vấn đề sẽ được thực hiện một cách nhanh chóng. Ngoài ra mã nguồn được xem lại liên tục và được chữa lỗi ngay trong quá trình phát triển.

Quyền sở hữu mã kết tập

Bất kỳ người nào trong đội dự án đều có quyền thay đổi mã trong quá trình làm việc với khách hàng chỉ cần tuân theo Tiêu chuẩn mã hoá và phải đảm bảo thực hiện thử nghiệm lại toàn bộ sau khi hoàn tất công việc sửa đổi. Điều này sẽ loại bỏ các vấn đề như là sai lệch về cấu trúc chương trình, … có thể xảy ra khi một cá nhân mã hoá độc lập.

Tái chế

Tái chế là kỹ thuật làm tăng hiệu quả của việc thết kế các mã có sẵn mà không làm thay đổi chức năng. Tái chế là rất khả thi vì một nhóm XP có các quy trình thử nghiệm tự động bắt lỗi, cho phép ta thay đổi mã (phản ánh khả năng hiểu bài toán ngày càng cao của các thành viên). Qua đó cũng mở rộng các thiết kế lên.

Tương tác liên tục

Các nhóm XP chia công việc ra thành các bước nhỏ và tích hợp mã của họ một vài lần trong một ngày. Do vậy, các vấn đề sẽ được xem xét ngay sau khi thực hiện và có thể dễ dàng sửa chữa khi gặp sự cố. Quá trình này đảm bảo cho mọi người luôn làm việc với phiên bản mới nhất của hệ thống.

Phát hành nhỏ gọn

Do nhóm XP làm việc trong các bước nhỏ cho nên việc phát hành cũng chia ra thành các phát hành nhỏ (khoảng vài tuần một lần). Và các thành viên sẽ phải tích hợp liên tục. Có những đề án XP thực hiện việc phát hành hàng ngày.

Khách hàng trực diện

Các lập trình viên phải luôn tiếp xúc với khách hàng để xác định rõ nhu cầu bất kể nỗ lực tốn bao nhiêu. Các nhà lập trình XP không nên suy đoán các vấn đề cụ thể của một chức năng mà phải hỏi trực tiếp khách hàng.

Trò chơi kế hoạch

Đây là một phần quan trọng và là chìa khóa quyết định tính chất và sự thành công của mô hình XP. Quá trình thực hiện Trò Chơi Kế Hoạch được thực hiện lặp đi lặp lại nhiều lần trong một dự án và diễn ra như sau: Để xác định yêu cầu được ưu tiên nhất (user story) và thời điểm gần nhất sắp tới sẽ bàn giao sản phẩm cho khách hàng (trong mẫu XP sản phẩm được chia thành rất nhiều phần nhỏ và được bàn giao nhiều lần) khách hàng và nhóm phát triển sẽ ngồi lại cùng nhau để nói chuyện. Khách hàng nêu ra những yêu cầu của mình cho nhóm phát triển trong đó có những người có kinh nghiệm phụ trách kỹ thuật của dự án. Nhóm phát triển sẽ chèo lái cho các yêu cầu trở nên rõ ràng ngay trên bàn họp, sau đó các yêu cầu sẽ được viết xuống thể câu chuyện. Sau khi có được các yêu cầu nhóm phát triển sẽ bắt đầu đo lường thời gian cho từng yêu cầu. Sau khi biết được ước lượng thời gian cho các yêu cầu thì khách hàng phải đưa ra được mức độ ưu tiên cho từng yêu cầu. Dựa vào hoàn cảnh của khách hàng và nhóm làm dự án (tiền của khách hàng, mức độ gấp gáp của khách hàng, nguồn lực của nhóm phát triển dự án, kỹ thuật để thực hiện các yêu cầu) khách hàng sẽ đưa ra những yêu cầu nào sẽ được thực thi trong lần bàn giao gần nhất. Thời điểm bàn giao gần nhất sẽ được cả hai bên thống nhất. Đó là phần đưa ra những yêu cầu và thời điểm bàn giao sản phẩm gần nhất. Còn một phần quan trọng nữa trong trò chơi kế hoạch là việc lên kế hoạch thực hiện các yêu cầu của nhóm làm dự án. Các thành viên phân chia cặp đôi và chọn cho mình những yêu cầu để phát triển, họ phải ước lượng lại thời gian thực hiện tác vụ đó.

Tóm lại Trò Chơi Kế Hoạch có 3 thao tác cơ bản là: thăm dò (lấy yêu cầu), thống nhất (thống nhất độ ưu tiên và thời hạn phát triển các yêu cầu) điều chỉnh (Nếu có sai sót trong quá trình đo lường hoặc khách hàng muốn thay đổi thì làm trong bước này)

Tuần lễ 40 giờ

Việc phát triển phần mềm là một công việc sáng tạo, và họ sẽ không thể sáng tạo được nếu họ kiệt sức. Việc giới hạn số giờ làm việc trong tuần sẽ đảm bảo được sức khoẻ của các thành viên và tăng cường chất lượng sản phẩm.

Ẩn Dụ

Dùng một hệ thống các thuật ngữ về đối tượng doanh nghiệp trong việc discuss các vấn đề và các giải pháp cho chúng (cái hệ thống Ẩn Dụ).

Dưới đây là một số nhận xét về XP và Rational Unified Process (RUP) Quá Trình Hợp Lý Thông Nhất.

Giống nhau

RUP không đề cập tỉ mỉ đến những định lý, tuy nhiên những nguyên lý cơ bản của phương pháp luận được đề ra rõ ràng và ta có thể thấy rõ, ví dụ như phản hồi từ khách hàng, sự thay đổi tăng dần, đầu tư ban đầu ít, thử nghiệm cụ thể và tuỳ biến theo từng nơi.

Có một số điểm tương đồng trong chiến lược lập kế hoạch, cả hai phương pháp đều phát biểu là không lập kế hoạch quá cụ thể ngay từ ban đầu bởi vì bạn không thể biết công việc gì là quan trong ngay từ lúc đầu. Cả hai phương pháp sử dụng quan niệm vòng quay của dự án, và nhấn mạnh sự ưu tiên theo mức độ quan trọng của các chức năng. Câu chuyện người sử dụng và trường hợp sử dụng đều được dùng để định hướng kiến trúc phần mềm và đóng vai trò quyết định cho việc lập kế hoạch cũng như quá trình phát triển.

Phương pháp luận hướng đối tượng đều là công cụ chính của cả hai phương pháp.

User story trong XP giống hệt với Trường Hợp Sử Dụng trong RUP.

Trong các yếu tố quan trọng cấu thành dự án là phạm vi dự án, tính đúng hạn, chất lượng và tài nguyên dự án, XP khuyến cáo chúng ta cần giữ cố định mục tiêu tính đúng hạn, chất lượng và tài nguyên. Phạm vi dự án có thể được phép điều chỉnh. RUP không đặt quan điểm một cách chặt chẽ như vậy, tuy nhiên RUP cho phép chúng ta dùng phương pháp xây dựng ma trận quyết định, trong đó các yếu tố đó có thể được điều chỉnh để phù hợp theo đặc tính của từng dự án. Dẫu RUP không phát biểu cụ thể như vây về sự cho phép thay đổi phạm vi dự án, tuy nhiên quan điểm tương đồng cũng thể hiện ở phương pháp phát triển phần mềm "to dần" của RUP.

Thử nghiệm cụ thể là một nguyên lý của XP và ta cũng có thể thấy ở trong RUP. Trong giai đoạn Khởi Thảo, RUP đòi hỏi bạn phải tiến hành nghiên cứu những vấn đề quan trọng và thử nghiệm những công nghệ mới.

Việc kiểm tra chương trình một cách tự động đều được XP và RUP khuyến cáo. XP dựa trên việc này để đảm bảo chi phí thấp cho mỗi sự thay đổi, tuy nhiên RUP thì không đòi hỏi.

Khác nhau

Sự khác biệt đáng kể giữa RUP và XP là RUP hướng đến những dự án lớn hơn so với XP và vì thế, nó phức tạp hơn.

Chi phí cho thay đổi và rủi ro

RUP cho rằng chi phí thay đổi tăng theo hàm mũ và tập trung vào cho những bước đầu tiên để giảm thiểu những chi phí về sau trong quá trình phát triển phần mềm. RUP quan tâm nhiều đến việc giảm thiểu rủi ro, theo dõi để tìm kiếm rủi ro và tiếp cận mục tiêu này từng bước từ quá trình xây dựng kiến trúc đến các quá trình phát triển phần mềm sau này.

XP cho rằng chi phí thay đổi không lớn lắm. XP cũng được xây dựng với mục tiêu giảm thiểu rủi ro (rủi ro ở đây định nghĩa là "những vấn đề cơ bản") và hướng tới một thiết kế và kiến trúc tốt cũng như với mục tiêu trước tiên là cho ra lò những chức năng quan trọng nhất. Tuy nhiên, với sự giả định là giá cho sự thay đổi thấp, kiến trúc của phần mềm được phép phát triển một cách hữu cơ, như một bộ xương chỉ cần phát triển vừa đủ để nâng đỡ cơ thể tại thời điểm nhất định. Điều then chốt trong XP là sự đơn giản.

Một đôi điều về tài liệu dự án

XP khuyến nghị bạn có một "hành lý" gọn nhẹ và không phải luôn luôn cập nhật tài liệu dự án, trừ khi bạn cần dùng chúng. RUP nhìn nhận vấn đề theo một cách khác và dựa trên sự cập nhật kịp thời bản thiết kế nhằm phục vụ cho việc phát triển phần mềm tiếp theo. XP cần có mã nguồn phải được viết tốt và dựa trên đó để trao đổi cũng như thiết kế cho các bước tiếp theo.

Chia sẻ mã nguồn

XP khuyến nghị cần có một sự chia sẻ mã nguồn trong nhóm, điều này có nghĩa là bất kỳ ai vào bất kỳ lúc nào cũng có thể thay đổi bất kỳ phần nào của mã nguồn. RUP gợi ý rằng người thiết kế phần mềm có trách nhiệm cho một phân hệ hoặc một gói nào đó, tương tự như vậy đối với lập trình viên.

Vị trí và trách nhiệm của các thành viên

Khi triển khai những công việc và vai trò trong dự án, RUP đồ sộ hơn nhiều so với XP. Những vai trò trong XP chỉ đơn thuần là Lập trình viên, Khách hàng, Huấn luyện viên và Quản lý. Đối với RUP thì ta cũng thấy LTV và KH, Coach tương tự như Kiến trúc sư và Quản lý tương tự như Lãnh đạo Chương Trình. Tuy nhiên, những trách nhiệm của các vị trí này không giống nhau bởi vì có một số hành động hoặc những concept không có trong XP. Những vị trí trong XP thường có nhiều trách nhiệm hơn so với RUP, ví dụ Lập Trình Viên là Khách Hàng, Khách hàng có trách nhiệm xây dựng yêu cầu.

Các bước của dự án

RUP có 4 bước được định nghĩa rõ ràng: Khởi động, Khởi Thảo, Xây dựng, and Chuyển Đổi. XP thì nói về Thăm Khảo, Cam Kết và Lái với những đầu mục công việc khác nhau và phân chia thời gian theo những phương pháp khác nhau. RUP chia toàn bộ quá trình sản xuất phần mềm thành một số phiên bản phát hành. XP chia và thực hiện dự án theo từng bước kế tiếp. Những bước này được XP gợi ý trong khoảng thời gian ngắn hơn.