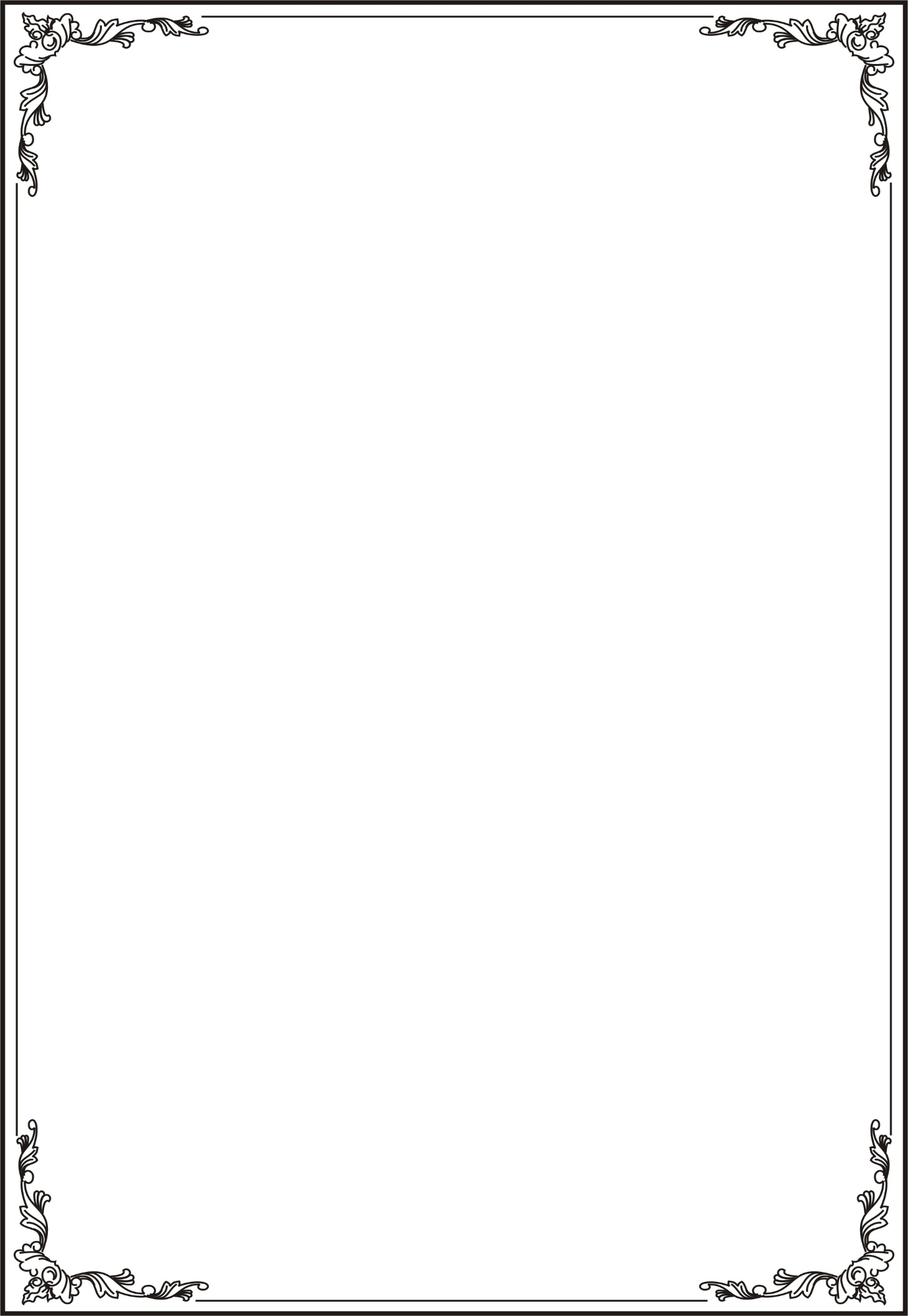
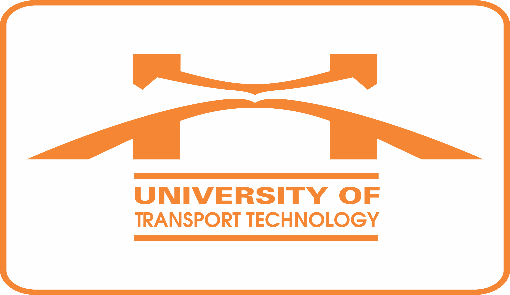
**ĐẠI HỌC CÔNG NGHỆ GIAO THÔNG VẬN TẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\*\*\*\*\*\*\***

****

**BÀI BÁO BÀI TẬP NHÓM MÔN HỌC**

**QUY TRÌNH VÀ CÔNG CỤ PHÁT TRIỂN PHẦN MỀM**

***Đề tài:* Chủ đề 8**

**Nhóm sinh viên thực hiện: Nhóm 1**

**Thành viên nhóm:** Nguyễn Văn Chung

Nguyễn Văn Tú

Nguyễn Văn Đạt

Dương Minh Đạt

Nguyễn Trọng Tấn

**Giảng viên phụ trách:** Bùi Thị Nhung

*Vĩnh Phúc, năm 2024*

**MỤC LỤC**

[**MỤC LỤC** 1](#_Toc167129460)

[**Chương 5: Quy trình phát triển phần mềmeXtreme Programming (XP)** 2](#_Toc167129461)

[5.1. Giới thiệu về XP 2](#_Toc167129462)

[5.2. Các giá trị cốt lõi của XP 2](#_Toc167129463)

[5.3. Vòng đời phát triển của một dự án XP 3](#_Toc167129464)

[5.4. Các công việc cốt lõi trong XP 4](#_Toc167129465)

# **Chương 5: Quy trình phát triển phần mềmeXtreme Programming (XP)**

## 5.1. Giới thiệu về XP

Lập trình cực độ (eXtreme Programming, viết tắt là XP) là một phương pháp linh hoạt dành cho các nhóm phát triển phần mềm nhỏ và trung bình xây dựng các phần mềm có yêu cầu thay đổi một cách nhanh chóng.

Đối với lập trình viên, XP đảm bảo rằng họ sẽ làm những công việc hữu ích theo khả năng của họ. Đối với khách hàng và người quản lý, XP đảm bảo rằng mang lại những lợi ích tốt nhất có thể sau mỗi thời gian làm việc. Họ sẽ nhìn thấy mục đích của quá trình thực hiện và có thể thay đổi nó mà không phát sinh nhiều chi phí. 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). XP giữ cho cái giá phải trả này ở mức thấp do vậy sẽ thúc đẩy môi trường sản xuất phần mềm.

## 5.2. Các giá trị cốt lõi của 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.

## 5.3. Vòng đời phát triển của một dự án XP

5.3.1. Khởi tạo (Exploration)  
 Khách hàng sẽ viết toàn bộ một Story Card mà họ hi vọng sẽ được thêm vào trong phiên bản đầu tiên. Mỗi Story Card mô tả chức năng được thêm vào trong chương trình. Tại cùng một thời điểm đó Project Team sẽ làm quen với Story Card bằng các Tools, Công nghệ và thực hành, họ sẽ sử dụng trong Project. Công nghệ sử dụng sẽ được kiểm tra kỹ lưỡng, còn kiến trúc hệ thống nếu như có triển vọng sẽ được khảo sát một cách tỉ mỉ bằng việc xây dựng các mẫu ban đầu. Thời gian để hoành thành pha này mất khoảng vài tuần cho tới vài tháng, điều đó còn phụ thuộc vào độ phức tạp của công nghệ đối với các Programmers.  
 5.3.2. Lập kế hoạch (Planning)

Pha này sẽ thiết lập các vị trí ưu tiên cho các Story. Ngoài ra còn xác nhận đồng ý cho nội dung của Version đầu tiên. Programmer trước tiên sẽ ước lượng độ yêu cầu của mỗi Story và lên kế hoạch làm việc sau thời điểm xác nhận đồng ý nội dung. Quãng thời gian của kế hoạch làm việc cho version đầu tiên không vượt quá 2 tuần  
  
 5.3.3. Chuyển giao từng phần (Iterations to Release)   
 Pha này bao gồm một vài bước lặp của hệ thống trước khi cho ra đời Version đầu tiên. Kế hoạch làm việc được thiết lập trong bản kế hoạch bị hỏng tới một số lượng bước lặp nào đó sẽ mất từ 1 tới 4 tuần để cài đặt. Bước lặp đầu tiên sẽ tạo hệ thống với kiến trúc của một hệ thống đầy đủ. Đó là bản lưu trữ bởi việc lựa chọn các story sẽ thúc đây việc xây dựng cấu trúc cho hệ thống đầy đủ. Khách hàng chấp nhận story được lựa chọn cho mỗi bước lặp. Quá trình test các chức năng sẽ được tạo bởi khách hàng được thực thi ở cuối mỗi bước lặp. Tại cuối bước lặp cuối cùng hệ thống đủ sẵn sang để có thể sản xuất.  
  
 5.3.4. Triển khai hoàn thiện sản phẩm (Productionizing)  
 Pha này yêu cầu mở rộng hơn về việc Testing cũng như Checking về khả năng thực thi của hệ thống trước khi hệ thống chính thức được giao tới tay của khách hàng. Tại pha này, sự thay đổi mới vẫn có thể được tìm ra và quyết đinh thực hiện chúng nếu như chúng được chứa trong phiên bản hiện tại. Trong suốt quá trình của pha này, các bước lặp có thể sẽ trở nên cần thiết để xử lí nhanh chóng từ 3 tới 1 tuần. Sau khi phiên bản được phát hành đầu tiên được sản xuất hóa cho khách hàng sử dụng, dự án XP phải giữ được hệ thống trong quá trình thực thi sản phẩm khi hầu hết việc thực hiện các bước lặp mới. Để làm được việc đó, thì pha Bảo trì cần sự nỗ lực từ việc hỗ trợ của khách hàng. Theo cách đó tốc độ phát triển có thể được hãm lại sau khi hệ thống đã trở thành sản phẩm.

5.3.5. Duy trì sản phầm (Maintenance)

Cũng gần giống như việc khi khách hàng không còn bất cứ 1 story nào để phát triển. Quy định này có nghĩa như việc khách hàng cần thỏa mãn hệ thống ở nhiều khía cạnh. Đây là thời điểm trong quá trình XP xử lý khi những tài liệu cần thiết của hệ thống được hoàn thành hay như việc không có bất kì thay đổi nào trong kiến trúc, thiết kế hoặc code. Death hầu hết xảy ra nếu như hệ thống không được giao đúng như yêu cầu hoặc nếu như nó trở nên quá đắt đỏ so với chi phí để phát triển.

## 5.4. Các công việc cốt lõi trong XP

5.4.1. Lập kế hoạch (The Planning Game)   
 Với XP, khách hàng tham gia trực tiếp vào quá trình lập kế hoạch phát triển phần mềm. Vai trò của khách hàng và nhóm phát triển được định ra một cách rõ ràng. Trách nhiệm của khách hàng:   
- Mô tả tính năng phần mềm cần phát triển thông qua các ' câu chuyện' (user story). User story có ý nghĩa tương tự như use case trong UML nhưng mức độ mô tả thì không chi tiết bằng. Phân loại các user story theo mức độ quan trọng từ quan điểm người sử dụng (dựa trên giá trị kinh doanh-business value). Từ đó sẽ định ra tính năng nào cần phải phát triển và phát triển theo thứ tự như thế nào. - Định ra thời điểm và chu kỳ bàn giao sản phẩm   
Trách nhiệm của nhóm phát triển:  
 - Ước lượng yêu cầu kỹ thuật (để phát triển) cho từng user story (ước lượng độ phức tạp).   
- Ước lượng thời gian, nhân công cũng như giá thành để phát triển từng user story.

5.4.2. Chuyển giao từng phần (Small releases)   
 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. Theo quy cách này, nhóm phát triển sẽ phát triển dần dần phần mềm, từ đơn giản đến phức tạp. Từng phần sẽ được chuyển giao cho khách hàng để có được ngay sự phản hồi của khách hàng. Từ đó sẽ có thể điều chỉnh ngay được sản phẩm cho phù hợp với yêu cầu của khách hàng. Khách hàng cũng có điều kiện để bổ sung hay thay đổi yêu cầu phần mềm.   
5.4.3. Bảng định danh (Metaphor)   
 Nhóm phát triển XP dùng chung một hệ thống các thuật ngữ để biểu diễn hệ thống cần phát triển. Các thuật ngữ này sẽ được dùng trong khi trao đổi giữa các thành viên trong nhóm cũng như khi trao đổi với khách hàng.

5.4.4. Thiết kế đơn giản (Simple design)   
 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.  
 XP khuyến khích tìm kiếm giải pháp đơn giản khi thiết kế phần mềm. Chỉ thiết kế phần mềm thoả mãn yêu cầu hiện tại của khách hàng, không nên tìm kiếm một giải pháp cho một hệ thống tương lai. Theo đó, chỉ cần một thiết kế làm sao cho chương trình chạy được và thỏa mãn yêu cầu của khách hàng.   
5.4.5. Kiểm thử liên tục (Testing)   
Các lập trình viên sẽ viết các đơn vị 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. XP yêu cầu rất cao trong khâu kiểm thử và kiểm định chương trình. Với mỗi phần của chương trình, lập trình viên phải viết chương trình kiểm thử cho phần đó trước khi thực sự bắt đầu khi viết chương trình (cho phần đó). Khách hàng sẽ chịu trách nhiệm thực hiện kiểm định sản phẩm.   
 5.4.6. Hoàn thiện liên tục (Refactoring)   
 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.   
Quan điểm của XP là chất lượng phần mềm được thể hiện bằng chất lượng của mã nguồn (code). Một chương trình được viết rõ ràng, đơn giản thì sẽ dễ bảo dưỡng và thay đổi. XP khuyến khích tổ chức lại chương trình một cách đều đặn để nâng cao tính sáng sủa của chương trình, dễ bổ sung các chức năng mới, nâng cao hiệu suất của chương trình.

5.4.7. Lập trình theo đôi (Pair programming)   
 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.   
Tất cả các phần chương trình do một hay nhiều nhóm hai người viết. Hai người này sẽ sử dụng chung một máy tính, cùng đồng thời viết chương trình. Quy cách này sẽ giúp cho có được giải pháp lập trình tốt hơn, chương trình sẽ có chất lượng và hiệu quả hơn.  
 5.4.8. Chia sẻ công việc (Collective ownership)  
 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.   
Tất cả mã nguồn đều thuộc quyền sở hữu của mọi thành viên trong nhóm phát triển. Theo đó, mã nguồn có thể được sửa đổi ngay khi cần. Với cách quản lý thông thường, mỗi phần mã nguồn thường do một người quản lý, nên khi cần sửa đổi thì phải cần sự thông qua chủ sở hữu, đôi khi điều này gây mất nhiều thời gian.   
5.4.9. Tích hợp liên tục (Continuous integration)  
 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.   
Việc tích hợp sẽ được tiến hành một cách liên tục. Khi một đoạn chương trình mới được phát triển, đã vượt qua phần kiểm thử, thì sẽ được tích hợp ngay vào hệ thống. Điều này sẽ giúp cho việc phát hiện và sửa lỗi thích hợp nhanh hơn và rẻ hơn. Trong một ngày có thể thực hiện nhiều lần tích hợp hệ thống.  
 5.4.10. Làm việc cùng khách hàng (On-site customer) 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.   
Với XP, khách hàng sẽ tham gia cách trực tiếp trong suốt quá trình phát triển phần mềm. Sự tham gia này sẽ giúp cho nhóm phát triển có điều kiện tham khảo trực tiếp ý kiến của khách hàng, trao đổi về hệ thống cần được phát triển, tránh được nhầm lẫn trong cách hiểu về hệ thống cần phát triển. Mục tiêu cuối cùng là sản phẩm làm ra phù hợp với yêu cầu của khách hàng.   
5.4.11. Sử dụng các chuẩn viết mã (Coding standards)   
 Đây là một loạt các quy ước về mã hoá để các thành viên của dự án theo đó làm. Để chương trình (mã nguồn) có thể hiểu được một cách dễ dàng, nhất là đối với các quy cách lập trình đôi và sở hữu tập thể, nhóm phát triển phải thống nhất cách viết chương trình. Cần phải có một quy định cụ thể, rõ ràng về cách viết (ví dụ, cách đặt tên biến, cách bổ sung chú thích, ..v.v.) để làm sao tất cả đều hiểu được.   
5.4.12. Giới hạn 40 giờ/tuần (40-hour week)  
 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.   
Hiện tượng làm việc quá giờ rất phổ biến trong giới phát triển phần mềm. Thực tế cho thấy khi người lao động làm việc quá giờ thường hay mệt mỏi, dẫn đến làm việc không hiệu quả, chất lượng sản phẩm giảm. XP khuyến cáo không nên làm việc quá giờ, chỉ làm đúng giờ quy định để đảm bảo chất lượng sản phẩm.