Kế hoạch đảm bảo

Chất lượng phần mềm

Cho

Website quản lý thông tin nhà trọ

Phiên bản 1.0 được phê chuẩn

Được chuẩn bị bởi

Nguyễn Chí Tâm MSSV 1111334

Nguyễn Hoài Nam MSSV 1111314

Nguyễn Quí Nghĩa MSSV 1111316

Quách Hoàng Phúc MSSV 1111326

Trần Văn Tùng MSSV 1111364

Võ Văn Hiệp MSSV 1111289

Nguyễn Phương Ghi MSSV 1111284

Huỳnh Mai Hoàng Huy MSSV 1111296

Trần Duy Lâm MSSV 1111305

Nhóm 2

7/11/2014

Contents

**No table of contents entries found.**

Theo dõi phiên bản

1. **Mục đích**

* Nhằm không ngừng nâng cao chất lượng sản phẩm Website quản lý thông tin phòng trọ sau khi phát hành

1. **Bảng chú giải thuật ngữ**
2. **Tài liệu tham khảo**

* Tài liệu đặc tả yêu cầu phần mềm Website quản lý thông tin nhà trọ - Phiên bản 1.0 – Ngày 5/9/2014 – Nhóm 2.
* Tài liệu thiết kế phần mềm Website quản lý thông tin nhà trọ - Phiên bản 1.3 – Ngày 17/09/2014 – Nhóm 2.

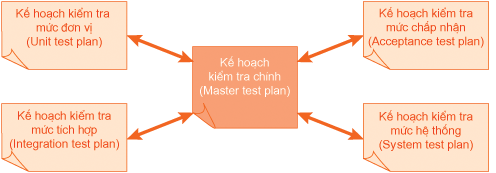
1. **Quản lý**
   1. **Tổ chức**
   2. **Công việc**
   3. **Vai trò và trách nhiệm**
   4. **Các nguồn tài nguyên được dự đoán để đảm bảo chất lượng**
2. **Tài liệu**
   1. **Mục đích**
   2. **Yêu cầu tài liệu tối thiểu**
      1. **Mô tả yêu cầu phần mềm**
      2. **Mô tả thiết kế phần mềm**
      3. **CÁc kế hoạch thẩm tra và công nhận hợp lệ**

Mục đích: Nhằm chỉ định và mô tả các loại kiểm tra sẽ được triển khai và thực hiện. Kết quả của bước lập kế hoạch là bản tài liệu kế hoạch KTPM, bao gồm nhiều chi tiết từ các loại kiểm tra, chiến lược kiểm tra, cho đến thời gian và phân định lực lượng kiểm tra viên.

Bản kế hoạch kiểm tra đầu tiên được phát triển rất sớm trong chu trình phát triển phần mềm (PTPM), ngay từ khi các yêu cầu đã tương đối đầy đủ, các chức năng và luồng dữ liệu chính đã được mô tả. Bản kế hoạch này có thể được coi là bản kế hoạch chính (master test plan), trong đó tất cả các kế hoạch chi tiết cho các mức kiểm tra và loại kiểm tra khác nhau đều được đề cập.

Lưu ý, tùy theo đặc trưng và độ phức tạp của mỗi dự án, các kế hoạch kiểm tra chi tiết có thể được gôm chung vào bản kế hoạch chính hoặc được phát triển riêng.

Sau khi bản kế hoạch chính được phát triển, các bản kế hoạch chi tiết lần lượt được thiết kế theo trình tự thời gian phát triển của dự án.



Bản kế hoạch chính và các bản kế hoạch chi tiết

Xác định yêu cầu kiểm tra: chỉ định bộ phận, thành phần của PM sẽ được kiểm tra, phạm vi hoặc giới hạn của việc kiểm tra. Yêu cầu kiểm tra cũng được dùng để xác định nhu cầu nhân lực.

Khảo sát rủi ro: Các rủi ro có khả năng xảy ra làm chậm hoặc cản trở quá trình cũng như chất lượng kiểm tra. Ví dụ: kỹ năng và kinh nghiệm của kiểm tra viên quá yếu, không hiểu rõ yêu cầu.

Xác định chiến lược kiểm tra: chỉ định phương pháp tiếp cận để thực hiện việc kiểm tra trên PM, chỉ định các kỹ thuật và công cụ hỗ trợ kiểm tra, chỉ định các phương pháp dùng để đánh giá chất lượng kiểm tra cũng như điều kiện để xác định thời gian kiểm tra.

Xác định nhân lực,vật lực: kỹ năng, kinh nghiệm của kiểm tra viên; phần cứng, phần mềm, công cụ, thiết bị giả lập… cần thiết cho việc kiểm tra.

Lập kế hoạch chi tiết: ước lượng thời gian, khối lượng công việc, xác định chi tiết các phần công việc, người thực hiện, thời gian tất cả các điểm mốc của quá trình kiểm tra.

Tổng hợp và tạo các bản kế hoạch kiểm tra: kế hoạch chung và kế hoạch chi tiết.

Xem xét các kế hoạch kiểm tra: phải có sự tham gia của tất cả những người có liên quan, kể cả trưởng dự án và có thể cả khách hàng. Việc xem xét nhằm bảo đảm các kế hoạch là khả thi, cũng như để phát hiện (và sữa chữa sau đó) các sai sót trong các bản kế hoạch.

* + 1. **Báo cáo các kết quả thẩm tra và công nhận hợp lệ**

Phân tích kết quả kiểm tra và đề xuất yêu cầu sửa chữa: Chỉ định và đánh giá sự khác biệt giữa kết quả mong chờ và kết quả kiểm tra thực tế, tổng hợp và gửi thông tin yêu cầu sửa chữa đến những người có trách nhiệm trong dự án, lưu trữ để kiểm tra sau đó.

 Đánh giá độ bao phủ: Xác định quá trình kiểm tra có đạt được độ bao phủ yêu cầu hay không, tỷ lệ yêu cầu đã được kiểm tra.

Phân tích lỗi: Đưa ra số liệu phục vụ cho việc cải tiến các qui trình phát triển, giảm sai sót cho các chu kỳ phát triển và kiểm tra sau đó. Ví dụ, tính toán tỷ lệ phát sinh lỗi, xu hướng gây ra lỗi, những lỗi “ngoan cố” hoặc thường xuyên tái xuất hiện.

Xác định quá trình kiểm tra có đạt yêu cầu hay không: Phân tích đánh giá để xem các Test Case và chiến lược kiểm tra đã thiết kế có bao phủ hết những điểm cần kiểm tra hay không? Kiểm tra có đạt yêu cầu dự án không? Từ những kết quả này, kiểm tra viên có thể sẽ phải thay đổi chiến lược hoặc cách thức kiểm tra.

* + 1. **Tài liệu người dùng**
    2. **Kế hoạch quản lý cấu hình phần mềm**
    3. **Các tài liệu khác**

1. **Các chuẩn, thực tiển, quy ước và các phép đó**
   1. **Mục đích**

Mục đích của việc đặt các chuẩn, quy ước đặt tên các hàm, biến trong lập trình là:

* Giúp cho các thành viên trong nhóm lập trình có thể dễ dàng hiểu, và sử dụng các hàm, biến của người người khác trong nhóm, để có thể giải quyết dễ dàng, nhanh chóng các bài toán khác nhau.
* Miêu tả một cách đầy đủ, rõ ràng các tiến trình của biến, hàm

Thường xuyên kiểm tra các hàm, các biến có đặt tên đúng quy cách hay không nhằm nhắc nhở kịp thời các thành viên tuân thủ đúng các quy chuẩn lập trình của nhóm lập trình.

* 1. **Nội dung**
* Quy ước đặt tên trong cơ sở dữ liệu: được viết hoa toàn bộ vầ không dấu, được ngăn cách bởi dấu “\_”.

Ví dụ: NHA\_TRO, DIA\_CHI,…

* Quy ước đặt tên hàm trong controller: viết thường chữ đầu tiên và viết hoa chữ cái đầu tiên kể từ chữ thứ hai trỡ đi
* Quy ước đặt tên các view, controler: chữ cái đầu tiên của tên đầu tiên viết thường và chữ cái đầu tiên của tên thứ hai viết hoa

Ví dụ: tinhThanh.php, binhLuan.php

* Tên biến phải mang ý nghĩa. Nên tránh các biến sử dụng một ký tự, trừ chúng là những biến thực sự không có ý nghĩa hoặc biến dùng tạm. VD: những biến cần mang ý nghĩa như: $nhaTro, $taiKhoan,…những biến không mang ý nghĩa hoặc dùng tạm như: $i, $n,…
* Các tài liệu được thực hiện trong dự án được đặt tên phải được bắt đầu “HTQLNT-”. VD: tài liệu đặc tả: HTQLNT – Đặc tả yêu cầu.docx,…
* Các thành viên trong nhóm mõi ngày phải update SVN ít nhất một lần và xem bảng phân công công việc và phải hoàn thành sớm các yêu cầu đúng tiến độ.

1. **Xem lại phần mềm**
   1. **Mục đích**

Xác định các xem lại phần mềm sẽ được thực hiện. Chúng có thể bao gồm các xem lại về quản lý, xem lại về kỹ thuật, thanh tra, kiểm toán.

* + - Liệt kê lịch biểu cho các xem lại phần mềm khi chúng có liên quan với lịch biểu của dự án phần mềm.
    - Trình bày cách thức các xem lại phần mềm sẽ được hoàn thành.
    - Trình bày các hoạt động thêm (sẽ được yêu cầu) và cách thức chúng sẽ được thực hiện và được kiểm tra.
  1. **Các yêu cầu tối thiểu**
     1. **Xem lại đặc tả phần mềm**

Đảm bảo sự đầy đủ của các yêu cầu chứa năng và phi chức năng được trình bày trong mô tả yêu cầu phần mềm:

* + - Kiểm tra sản phẩm có đầy đủ các tính năng và hoạt động đúng với đặt tả không.
    - Kiểm tra các yêu cầu phi chức năng được đáp ứng đến mức độ nào.
    1. **Xem lại thiết kế kiến trúc**

Thực hiện xem lại thiết kế kiến trúc phần mềm để đánh giá sự đầy đủ mức cao của sản phẩm như được viết trong mô tả thiết kế phần mềm.

* + - Sản phẩm có tuân thủ kiến trúc MVC, mô hình 3 lớp.
    1. **Xem lại thiết kế chi tiết**

Xác định tính có thể chấp nhận của các thiết kế phần mềm chi tiết.

* + - Đảm bảo rằng hệ thống đầy đủ các chức năng như trong mô hình phân rã (DFD) của hệ thống và hoạt động chính xác với mô tả chức năng của hệ thống.
    - Kiểm tra cơ sở dữ liệu và các ràng buộc dữ liệu có đúng với dữ liệu đã thiết kế.
    1. **Xem lại kế hoạch thẩm tra và công nhận hợp lệ**

Đánh giá sự hoàn chỉnh và đầy đủ của các phương pháp thẩm tra và công nhận hợp lệ đã được định nghĩa trong các kế hoạch thẩm tra và công nhận hợp lệ.

* + - Sản phẩm có vượt qua tất cả các trường hợp kiểm thử không.
    - Dùng checkstyle kiểm tra xem có lỗi trình bài không.
    1. **Kiểm toán chức năng**

Trước khi phát hành phần mềm phải kiểm tra rằng tất cả các yêu cầu được xác định trong mô tả yêu cầu phần mềm đã được đáp ứng. Các quyền của từng loại người dùng có được đáp ứng đầy đủ không.

* + 1. **Kiểm toán vật lý**

Thực hiện kiểm tra phiên bản cuối cùng trước khi phát hành với chương trình và tài liệu:

* + - Kiểm tra các chức năng có đúng với tài liệu mô tả
    - Kiểm tra giao diện có phù hợp với mô tả trong tài liệu

Kiểm tra khả năng đáp ứng nhu cầu của người dùng.

* + 1. **Kiểm toán trong quy trình**

Kiểm tra các mã lệnh có nhất quán với các quy ước đã được mô tả trong tài liệu

Kiểm tra giao diện phần mềm

Kiểm tra thực hiện các thiết kế không nhất quán với yêu cầu chức năng.

Thực hiện các yêu cầu chức năng không nhất quán với các mô tả kiểm thử

* + 1. **Xem lại về quản lý**

Định kỳ thực hiện đánh giá sự thực thi của các hoạt động và các thành phần được mô tả trong tài liệu.

Thực hiện xem lại về công tác nhân sự của dự án.

* + 1. **Xem lại kế hoạch quản lý cấu hình**

Định kỳ thực hiện đánh giá hoạt động quản lý cấu hình được định nghĩa trong kế hoạch quản lý cấu hình

* + 1. **Xem lại sau thực thi**

Thực hiện sau cùng trong quy trình phần mềm nhằm đánh giá các hoạt động phát triển của dự án. Cung cấp các đề xuất cho các hoạt động phù hợp để phát hành sản phẩm

* + 1. **Các kiểm toán và xem lại khác.**

Xem lại tài liệu người dùng:

* + - Tính đầy đủ
    - Tính rõ ràng
    - Tính chính xác
    - Tính tiện lợi

1. **Kiểm thử**

* Kiểm thử chức năng cửa phần mềm:
  + Phân tích mã nguồn với công cụ hỗ trợ lập trình web: notepad++, adobe dreamweaver, kiểm tra các function, viết trường hợp kiểm thử và tài liệu kết quả.
  + Kiểm thử chức năng để kiểm tra sự thực thi của các chức năng trên nền tảng hệ điều hành Windows 8, viết trường hợp kiêm thử và viết tài liệu kết quả.
* Kiểm thử giao diện:
  + Phân tích bằng thiết kế giao diện, nguyên tắc hoạt động, kiểm tra có lỗi hoạt động hay về mặt logic hoạt động của giao diện, thử nghiệm giao diện thực tế và viết tài liệu kết quả.
  + Sử dụng chương trình trong thời gian cụ thể, sử dụng các điều khiển trên giao diện để tìm lỗi, viết trường hợp kiểm thử và tài liệu kết quả.
* Kiểm thử các yêu cầu phi chức năng:
  + Sử dụng các chức năng, ghi nhận thời gian phản ứng, viết tài liệu kết quả.
  + Kiểm tra tương tác, kiểm tra các điều kiện sử dụng chức năng chương trình, kiểm tra dữ liệu, viết tài liệu kết quả.
  + Chạy phần mềm trên nền tảng hệ điều hành khác nhau để tìm ra những chức năng có khả năng gây ra lỗi.
  + Thử nghiệm đóng góp phần mềm, tiến hành cài đặt và chạy ứng dụng thực tế cho nhiều người sử dụng để đánh giá khả năng ứng dụng của phần mềm.

1. **Hoạt động hiểu chỉnh và báo cáo vấn đề**

* Các thực tiễn, các thủ tục sẽ tuân theo:
* Kiểm tra quá trình phát triển phần mềm để đảm bảo rằng các thủ tục và các chuẩn đảm bảo chất lượng được tuân theo
* Việc kiểm tra các chuẩn dự án được tuân theo, ứng dụng và các tài liệu được làm phù hợp với các quy định, các chuẩn đã thảo luận và thống nhất
* Trách nhiệm của nhóm quản lý chât lượng:
* Bảo đảm tất cả các thay đổi là được các bộ phận liên quan (nhóm lập trình, nhỏm bảo trì, nhóm kiểm thử) nhận biết và tham gia
* Xem xét, phê chuẩn hoặc từ chối các thay đổi
* Kiểm tra, xác nhận các thay đổi
* Phê chuẩn các bản phân phối sản phẩm đến khách hàng

1. **Công cụ, kỹ thuật và phương pháp**

Các công cụ phần mềm:

* + Tortoise SVN: giúp quản lý tài liệu, quản lý source code của dự án.
  + NetBean: trình soạn thảo source code của dự án.
  + Notepad++: trình soạn thảo source code của dự án.
  + Microsoft Word: trình soạn thảo tài liệu của dự án.
  + Google Chrome: trình duyệt web, môi trường làm việc và kiểm thử của sản phẩm.
  + FireFox: trình duyệt web, môi trường làm việc và kiểm thử của sản phẩm.
  + Paint: Công cụ hổ trợ thiết kế giao diện cho dự án.
  + PowerDesigner: hổ trợ thiết kế CSDL cho dự án.
  + SQL Server Studio: cấu hình CSDL cho dự án.
  + XAMPP: chương trình tạo máy chủ web, tạo môi trường làm việc và kiểm thử sản phẩm

Các kỹ thuật:

* + CodeIgniter: phát triển web động với mô hình MVC (model-view-controller)

1. **Kiểm soát phương tiện truyền thông**

* Nhận dạng phương tiện truyền thông cho từng sản phẩm trung gian và sản phẩm bàn giao (được làm ra do sử dụng máy tính) và các tài liệu cần được lưu trữ truyền thông, gồm quy trình phục hồi và sao chép.
* Bảo vệ phương tiện truyền thông vật lý của chương trình máy tính khỏi các truy xuất không được phép hay các hư hại vô ý hay sự suy thoái trong suốt các giai đoạn trong chu kỳ sống của phần mềm.

1. **Kiểm soát nhà cung cấp**

* Phát hành giấy phép sử dụng phần mềm cho các nhà cung cấp.
* Cung cấp giấy phép phát triển mã nguồn mở cho các nhà phát triển khác.

1. **Tập hợp hồ sơ, bảo trì và tiếp tục sử dụng**

Tài liệu của dự án sẽ được lưu trữ bằng công cụ quản lý cấu hình Tortoise SVN. Được tập hợp, quản lý, sắp xếp, bảo vệ và duy trì trong thời gian dài.

1. **Huấn luyện**

* Huấn luyện công nghệ.
* Huấn luyện kỹ năng nhóm.

1. **Quản lý rủi ro**

**15.1 Rủi ro:**

* Khái niệm về rủi ro: rủi ro là 1 hay nhiều sự việc chưa nhưng có khả năng xảy ra trong tương lai có tác động đến dự án, và khi sự việc đó xảu ra thường sẽ gây ảnh hường xấu tới dự án, cản trở sự hoàn thành của dự án.
* Rủi ro có 2 thuộc tính chủ yế là xác suất rủi ro sẽ xuất hiện và tác động của rủi ro nếu xuất hiện.
* Xác suất rủi ro xuất hiện: chúng ta có thế dùng tỉ lệ 0 – 8 để mô tả xác suất của rủi ro. Rủi ro có xác suất 0 được gọi là không có cơ hội xuất hiện. Rủi ro có xác suất 9 được gọi là chắc chắn xảy ra. Xác suất trong khoảng 0 – 8 thì rủi ro có coi hội suất hiện.
* Tác động của rủi ro nếu có xuất hiện: chúng ta có thể dung thang 0-8 để mô tả tác động của rủi ro. Rủi ro với tác động 0 được gọi là không có tác động. Rủi ro với tác động 8 được gọi là đình chỉ ( nguy hiểm nghiêm trọng dẫn đến dự án không thực hiện được).
* Nhận diện các rủi ro:

|  |  |
| --- | --- |
| Thành phần gây ra rủi ro | Quản lý rủi ro |
| Ghi đề lên công việc của người khác, đoạn mã không có trong phiên bản mới nhất. | Sử dụng công cụ quản lý cấu hình hiệu quả. |
| Thiếu thời gian thử nghiệm hoặc chưa biết cách sử dụng các sản phẩm công nghệ. | Dành thêm thời gian để học cách sử dụng các công cụ và nhưng công nghệ mới, tìm kiếm sự giúp đỡ từ giáo viên, đồng nghiệp. |
| Công việc bị chôn vùi trong công việc của nhóm khác | Phải có 1 kế hoạch quản lý xác định, thường xuyên cập nhật kế hoạch của nhóm. |
| Thiếu sự liên lạc giữa các nhóm | Thiết lập 1 trang web nhóm, tài khoản email cho nhóm, thường xuyên họp nhóm. |
| Sự tổ chúc cho dự án | Xác định vai trò cho các thành viên trong nhón |
| Khó khăn trong việc hợp nhất các công việc | Gia tăng giao tiếp, hợp nhất thường xuyên |
| Kế hoạch phát khai mất nhiều thời gian, không đủ thời gian để hoàn thành sản phẩm | Không đi sâu vào các chi tiết không cần thiết cho kế hoạch |
| Thiếu người phát triển | Sử dụng nhưng người tốt nhất, xây dựng nhóm làm việc, đào tạo người mới |
| Kế hoạch, dự toán không sát thực tế | Ước lượng bàng các phương pháp khác nhau lọc, loại bỏ các yêu cầu không quan trọng. |
| Phát triển sai chức năng | Chọn phương pháp phân tích tốt hơn, phân tích tính tổ chức/ mô hình nghiệp vụ của khách hàng |
| Phát triển sai giao diện | Phân tích tao tác người dùng, tạo kịch bản các dùng, tạo bản mẫu |
| Yêu cầu quá cao | Lọc bớt yêu cầu, phân tích chi phí/ lợi ích. |

* 1. Quản lý rủi ro:
* Trong các dự án công nghệ thông tin, tỉ lệ thành công theo nghĩa đạt được yêu cầu chất lượng, đúng hạn và không vượt chi là rất không cao, nguyên nhân chủ yếu là do không có, hoặc thực hiện không tốt việc phòng người và xử lý các nguye cơ dẫn đến thất bại của 1 dự án. Như vậy quản lý rủi ro có vai trò khá quan trọng trong toàn bộ tiến trình quản lý dự án. Mộ các hiều đơn giản, quản lý rủi ro là cách để quản lý các rủi ro, để làm giảm những tác động của những sự kiện không mong muốn phát sinh trong dự án.
* Tầm quan trọng của quản lý rủi ro: quản lý rủi ro là 1 nghệ thuậ và những nhận biết khoa học, là nhiệm vụ và sự đối phó rủi ro thông qua hoạt động của dự án và những mục tiêu đòi hỏi quan trọng nhất của dự án.

Quản lý rủi ro thường không được chú ý nhiều trong dự án, nhưng nó lại giúp cải thiện được sự thành công của dự án trong việc giúp chọn lựa nhứng dự án tốt, xác định phạm vi dự án và phát triển những ước tỉnh có tính thực tế.

* Mục đích của quản lý rủi ro trong công nghệ phần mềm: Quản lý rủi ro giúp cho 1 dự án tránh khỏi sự thất bại như không hoàn thành dự án như kế hoạch đã đinh, vượt quá ngân sách và không đáp ứng được sự mong đợi của khách hàng. Quản lý rủi ro tìm kiếm và xem xét từ các góc cạnh khác nhau trong các dự án để đảm bảo bằng những mối đe dọa cho các dự án được xác định và phân tích, tiến hành các chiến lược thích hợp để giảm nhẹ và khống chế rủi ro. Chúc năng chính của quản lý rủi ro là đoán nhận được tất cả nhưng rủi ro có khả năng ảnh hưởng đến 1 dự án, đánh giá mức độ nghiêm trọng và hậu quả, sau đó xác định các giải pháp tùy theo tính chất của các rủi ro. Giảo thiểu tối đa các yếu tốt bất ngờ và các vấn đề không mong đợi phát sinh trong suốt quá trình thực hiện dự án, bằng cách thiết lập ra các kế hoạch cho các tình huống có thể xảy ra. Nhưng kế hoạch này sẽ giảm thiểu tối đa nhưng tình huống có thể dẫn tới các sản phẩm lệch lạc hoặc có thể phá hủy toàn bộ dự án.
* Quản lý rủi ro làm giảm thiều khả năng rủi ro, trong khi đó tăng tối đa nhưng cơ hội tiềm năng.

Trước khi áp dụng bất cứ 1 quá trình quản lý rủi ro nào, các thành viên trong nhóm thực hiện dự án nên nắm được rõ ràng về các hậu quả sau này của các rủi ro trong dự án của họ như:

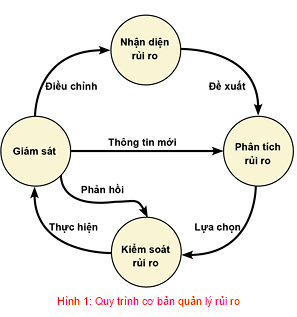
* Sự mất mát sẽ phát sinh nếu xuất hiện rủi ro: sự mất mát trong dự án phần mềm có thể đến như lợi nhuận, thì phần, khách hàng.
* Tính nghiêm trọng của sự mất mát
* Tính lâu dài của các rủi ro

Những mô hình quản lý rủi ro phần mềm phổ biến:

* Mô hình quản lý rủi ro của Boehm (win-win)
* Mô hình quản lý rủi ro phần mềm của SEI
* Mô hình quản lý rủi ro của Hall
* Mô hình quản lý rủi ro của Karolak
* Phương pháp luận rủi ro của Kontio

2. Quy trình quản lý rủi ro:

Nhận diện và kiểm soát tốt rủi ro chỉ bằng những kỹ năng và kinh nghiệm cá nhân không thì chưa đủ, việc kiểm soát rủi ro phải được thực hiện theo 1 quy trình chặt chẽ và phù hợp với đặc thù, mục tiêu và ngân sách của dự án.



15.2.1 Nhận diện rủi ro:

Xác định được chính xác các nguồn có khả năng phát sinh rủi ro là điều không dễ dàng. Thông thường rủi ro xuất hiện từ các nguồn sau:

* + - Ngân sách - nguồn tài trợ cho dự án
    - Thời gian thực hiện dự án
    - Thay đổi về phạm vi và yêu cầu dự án
    - Khó khăn về kỹ thuật
    - Hợp đồng giữa các bên
    - Môi trường, luật pháp, chính trị, văn hóa

Để nhận diện được rủi ro có nhiều kỹ thuật được áp dung. Các kỹ thuật này giúp cho dự án khoanh vùng và xác định dấu hiệu xuất hiện rủi ro, vừa giúp trảnh bỏ xót các dấu hiệu, vừa làm tăng kết quả và độ tin cật của việc nhận diện rủi ro. Từng kỹ thuật viên đề có nhưng hạn chế riêng, do đó việc kết hợp các kỹ thuật để có kết quả tốt nhất là cần thiết. Các kỹ thauatj được sử dung rộng rãi bao gồm:

* + - Xem xét tài liệu
    - Động não
    - Kỹ thuật Delphi
    - Nhóm danh nghĩa
    - Hỏi ý kiến chuyên gia
    - Sủ dụng phiếu kiểm tra hoặc bảng câu hỏi
    - Sử dụng biểu đồ
    1. Phân tích và phân loại rủi ro:
* Phân tích khả năng xuất hiện của rủi ro: có 4 mức để đo lường khả năng xuất hiện của rui ro, mỗi mức được gán với 1 giá trị số để có thể ước lượng sự quan trọng của nó:
* 6 - Thường xuyên: khả năng xuất hiện rủi ro rất cao, xuất hiện trong hầu hết dự án.
* 4 – Hay xảy ra: khả năng xuất hiện rủi ro cao, xuất hiện trong nhiều dự án.
* 2 – Đôi khi: khả năng xuất hiện rủi ro trung bình, chỉ xuất hiện ở 1 số ít dự án
* 1 - Hiếm khi: khả năng xuất hiện thấp, chỉ xuất hiện trong những điều kiện nhất định.
* Phân tích mức tác động của rủi ro:

Có 4 mức tác động của rui ro, mỗi mực độ được gán với 1 giá trị số để có thể ước lượng sự tác động của nó:

* 8 - Trầm trong: có khả năng làm dự án thất bại rất cao
* 6 – Quan trọng: gây khó khăn lớn và làm dự án không đạt được các mục tiêu
* 2 -Vừa phải: gây khó khăn cho dự án, ảnh hường việc đạt các mục tiêu của dự án
* 1 – Không đáng kể: gây khó khăn không đáng kể.
* Phân tích thời điểm xuất hiện rui ro:

Có 4 mực để ước lượng thời điểm rủi ro xuất hiện, mỗi mức được gán với 1 giá trị số để có thể ước lượng sự tác động của nó.

* 6 – Ngay lập tức: rủi ro xuất hiện gần như tức khắc
* 4 - Rất gần: rủi ro sẽ xuất hiện trong thời điểm rất gần thời điểm phân tích
* 2 - Sắp xảy ra: rủi ro sẽ xuất hiện trong tương lai gần
* 1 - Rất lâu: rủi ro sẽ xuất hiện trong tương lai xa hoăc chưa định được.
* Ước lượng và phân hạng các rủi ro:

Rủi ro được tính giá trị để ượng lượng bằng công thức:

Risk Exposure = Risk Impact \* Risk Probability \* Time Frame

Tiếp theo rủi ro được phân hạng từ cao đến thấp dựa theo các giá trị Risk Exposure tính toán được. Tùy theo tổ chức và đặc thù từng dự án, trưởng dự án sẽ xác định những rủi ro nào cần đưa vào kiểm soát, với các mức ưu tiên khác nhau.

15.2.3 Kiểm soát rủi ro:

Kiểm soát rủi ro bắt đầu với việc chọn lựa chiến lược và phương pháp đối phó rủi ro. Có nhiều chiến lược và phương pháp đối phó khác nhau, tùy theo tình huống dự án, môi trường và đặc thù của từng rủi ro. Trong thực tế các chiến lược phổ biếng bao gồm:

* + - * Tránh né
      * Chuyển giao
      * Giảm nhẹ
      * Chấp nhận
      * Sử dụng cây quyết định.

15.2.4 Giám sát và điều chỉnh:

Bao gồm hoạt động giám sát để bảo đảm các chiến lược đối phó rủi ro được lên kế hoạch và thực thi chặt chẽ. Việc giám sát cũng nhằm mục đích điều chỉnh các chiến lược hoặc kế hoạch đối phó nếu chúng tỏ ra không hiệu quả, không khả thi, ngốn quá nhiều ngân sách, hoặc để đáp ứng với rủi ro mới xuất hiện, hoặc sự biến tướng của rủi ro đã được nhận diện trước đó.

Kết quả giám sát có thể được báo cáo định kỳ đến tất cả những người có liên quan, đến quản lý cấp cao, hoặc đến khách hàng nếu cần thiết.

Trong thực tế, do các yếu tố liên quan đến dự án thay đổi liên tục, chu trình quản lý rủi ro không đi theo đường thẳng mà được lặp lại và điều chỉnh liên tục giữa các chặng. Các rủi ro liên tục được điều chỉnh hoặc nhận diện mới, do đó các chiến lược và kế hoạch đối phó cũng luôn được thay đổi để bảo đảm chúng khả thi và có hiệu quả.