

## Chương II

### TIỀN DỰ ÁN VÀ VÒNG ĐỜI DỰ ÁN (P1)

## Nội dung

- ❑ Tiền dự án
  1. Rà soát hợp đồng
  2. Kế hoạch phát triển và kế hoạch chất lượng
- ❑ Vòng đời dự án
  3. Giai đoạn phát triển
  4. Giai đoạn bảo trì

## II.1 Rà soát hợp đồng

- ❑ Các bước rà soát hợp đồng
- ❑ Mục tiêu rà soát
- ❑ Thực hiện rà soát

## Các bước rà soát hợp đồng

- ❑ Bước 1: Rà soát dự thảo đề cương (Proposal)
  - Rà soát dự thảo đề cương trước khi trình bày cho KH
  - Rà soát các tài liệu làm cơ sở cho đề cương
    - Tài liệu yêu cầu của KH
    - Dự toán chi phí và nguồn lực
    - Hợp đồng với đối tác (partners hoặc subcontractors)
- ❑ Bước 2: Rà soát dự thảo hợp đồng

## Mục tiêu rà soát dự thảo đề cương

- ❑ Đảm bảo các hoạt động sau đã được thực hiện thỏa đáng:
  1. Làm rõ và ghi chép đầy đủ các yêu cầu của KH
  2. Đánh giá năng lực của các đối tác với công ty thực hiện dự án
  3. Xác định một số vấn đề giữa KH và công ty như kênh liên lạc, cách thức chuyển giao, quy trình kiểm duyệt các pha chính, thủ tục yêu cầu thay đổi của KH

## Mục tiêu rà soát dự thảo đề cương

4. Đánh giá rủi ro phát triển PM
5. Dự toán đầy đủ nguồn lực dự án và thời gian biểu
6. Đánh giá năng lực của đội ngũ nhân viên
7. Đánh giá năng lực của KH
8. Xác định điều kiện tham gia của bên đối tác: đảm bảo chất lượng, tiến độ thanh toán, mức thù lao,...
9. Xác định và bảo vệ quyền sở hữu

## Mục tiêu rà soát dự thảo hợp đồng

- ❑ Đảm bảo các hoạt động sau đã được thực hiện thỏa đáng:
  1. Không tồn tại những vấn đề mơ hồ trong dự thảo hợp đồng
  2. Tất cả các thỏa thuận giữa KH và công ty đều được ghi đầy đủ và chính xác trong hợp đồng và phụ lục
  3. Không có những thay đổi, bổ sung, và thiếu sót mới nào được đưa vào dự thảo

## Thực hiện rà soát hợp đồng

- ❑ Những nhân tố ảnh hưởng đến mức độ nỗ lực rà soát:
  - Tâm của dự án
  - Độ phức tạp về kĩ thuật
  - Kinh nghiệm của nhân viên về lĩnh vực của dự án
- ❑ Ai có thể thực hiện rà soát?
  - Leader hoặc thành viên của đội đề xuất
  - Tất cả thành viên của đội đề xuất
  - Chuyên gia bên ngoài hoặc người của công ty không nằm trong đội đề xuất
  - Đội ngũ chuyên gia bên ngoài

## II.2 Kế hoạch phát triển và chất lượng

- ❑ Kế hoạch phát triển
- ❑ Kế hoạch chất lượng

## Kế hoạch phát triển

- ❑ Gồm các thành phần:
  - Sản phẩm của dự án
  - Giao tiếp dự án (Project interfaces)
  - Phương pháp và công cụ phát triển
  - Các thủ tục và chuẩn phát triển PM
  - Mô tả quy trình phát triển PM
  - Cột mốc(milestones)
  - Tổ chức đội ngũ
  - Phương tiện phát triển
  - Rủi ro phát triển
  - Phương thức kiểm soát việc thực hiện dự án
  - Ước tính chi phí cho dự án

## Kế hoạch chất lượng

- ❑ Gồm các thành phần:
  - Mục tiêu chất lượng
  - Các hoạt động rà soát
  - Kế hoạch kiểm thử
  - Công cụ và thủ tục quản lí cấu hình

## Thảo luận

- ❑ Rủi ro phát triển phần mềm?

## Rủi ro phát triển phần mềm

Nhóm rủi ro	Mục rủi ro
1. Quản lý nhân sự	1. Thiếu nhân sự có năng lực
2. Thời gian và ngân sách	2. Thời gian và ngân sách không thực tế
3. Chức năng của hệ thống	3. Phát triển chức năng không đúng 4. Phát triển giao diện người dùng sai
4. Quản lý yêu cầu	5. Bỏ sung thêm những yêu cầu không cần thiết 6. Sự thay đổi các tính năng của hệ thống không kiểm soát được
5. Hợp đồng phụ	7. Chất lượng kém từ những thành phần của hệ thống do đối tác thực hiện
6. Hiệu suất và sử dụng tài nguyên	8. Hiệu suất của tài nguyên kém 9. Các giải pháp về kỹ thuật chưa đủ để thực hiện các yêu cầu

13

## Những hoạt động quản lý rủi ro(RMA)

### □ RMA nội bộ

1. Phân tích chi tiết các yêu cầu, lịch trình, và ngân sách đã ước tính
2. Tổ chức hiệu quả về đội ngũ nhân sự
3. Đào tạo nhân sự, chuẩn bị nguồn nhân lực dự phòng
4. Sắp xếp user tham gia vào quy trình phát triển PM
5. Áp dụng phương pháp kiểm soát thay đổi hiệu quả
6. Áp dụng đo lường SQA

Đảm bảo chất lượng phần mềm

14

## Những hoạt động quản lý rủi ro(RMA)

### □ RMA nội bộ

7. Kiểm tra định kỳ sự sẵn sàng của các chuyên gia đang bận rộn với các dự án khác
8. Sắp xếp nhân sự có kinh nghiệm với rủi ro PM
9. Lập lịch cho các hoạt động giải quyết rủi ro sớm nhất có thể
10. Mô phỏng các tình huống rủi ro và chuẩn bị các giải pháp

Đảm bảo chất lượng phần mềm

15

## Những hoạt động quản lý rủi ro(RMA)

### □ RMA hợp đồng phụ

- Chuẩn bị và rà soát kỹ lưỡng hợp đồng phụ
- Kiểm soát tiến trình và thực hiện các hoạt động SQA đối với hợp đồng phụ
- Thuê cố vấn hỗ trợ đội thiếu kinh nghiệm

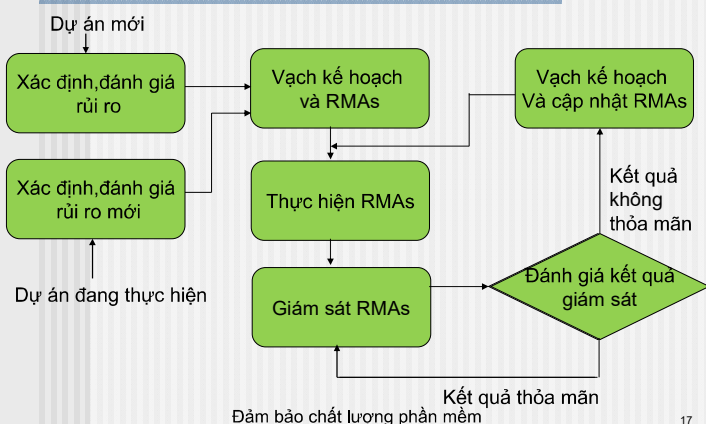
### □ RMA liên quan tới khách hàng

- Xây dựng và rà soát hợp đồng với KH
- Đàm phán với khách hàng để thay đổi những yêu cầu hoặc lịch biểu có rủi ro cao

Đảm bảo chất lượng phần mềm

16

## Quy trình quản lý rủi ro



17

## II.3 Giai đoạn phát triển

### □ Nội dung:

1. Các hoạt động SQA
2. Các yếu tố ảnh hưởng đến cường độ của các hoạt động SQA trong quy trình phát triển PM
3. Rà soát phần mềm
4. Kiểm thử phần mềm

Đảm bảo chất lượng phần mềm

18



### II.3.1 Các hoạt động SQA

Giai đoạn phát triển sẽ bao gồm các hoạt động SQA:

- Rà soát (Review)
  - Rà soát đặc tả yêu cầu, thiết kế, code, tài liệu
- Kiểm thử (Testing)
  - Kiểm thử đơn vị, tích hợp, hệ thống, chấp nhận

### II.3.2 Các yếu tố ảnh hưởng đến cường độ của các hoạt động SQA

#### □ Yếu tố “project”

- Tầm quan trọng của dự án
- Sự phức tạp kỹ thuật
- Quy mô của các thành phần trong PM có thể tái sử dụng
- Mức độ nghiêm trọng của các kết quả bị lỗi nếu dự án bị fail

### II.3.2 Các yếu tố ảnh hưởng đến cường độ của các hoạt động SQA

#### □ Yếu tố “team”:

- Trình độ chuyên môn của các thành viên trong nhóm
- Kinh nghiệm của nhóm trong lĩnh vực của dự án
- Sự sẵn có các nhân sự có thể hỗ trợ chuyên môn cho nhóm
- Tỷ lệ thành viên mới trong đội

### Ví dụ:

Một nhóm phát triển PM đã lên plan các hoạt động SQA cho dự án cấu trúc bộ người tiêu dùng mới. Hợp đồng ký với một cửa hàng đồ nội thất hàng đầu. Lĩnh vực của dự án đã được nhóm thực hiện hơn **10 lần trong 3 năm qua**.

Nhóm ước tính rằng khoảng 7 man-months cần giao cho 2 thành viên trong nhóm thực hiện, với thời gian khoảng 4 tháng. Nhóm ước tính **90% các thành phần trong PM có thể tái sử dụng** từ thư viện.

### Ví dụ:

STT	Hoạt động SQA	Thời gian thực hiện hoạt động SQA(ngày)	Thời gian điều chỉnh(ngày)
1	Rà soát yêu cầu	0.5	1
2	Kiểm tra thiết kế	1	1
3	Kiểm thử hệ thống PM đã làm hoàn chỉnh	4	2

### Verification và Validation

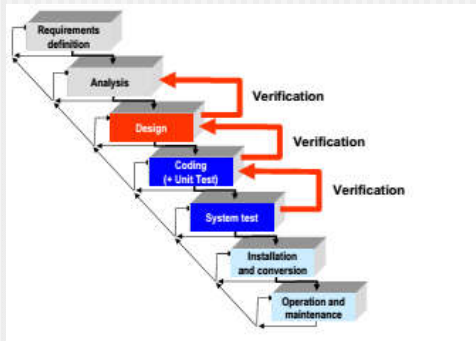
#### □ Verification(xác minh)

Quá trình kiểm tra PM để xác định các sản phẩm ở mỗi pha có thỏa các điều kiện được đặt ra ở đầu của pha đó.

#### □ Validation(Thẩm định)

Quá trình kiểm tra PM có đáp ứng được yêu cầu người dùng không.

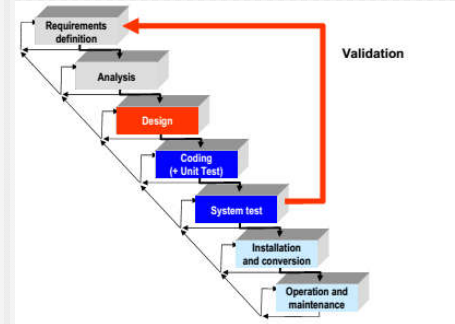
## Verification và Validation



Đảm bảo chất lượng phần mềm

25

## Verification và Validation



Đảm bảo chất lượng phần mềm

26

## II.3.3 Rà soát phần mềm

- ❑ Quá trình rà soát
- ❑ Mục tiêu rà soát
- ❑ Các phương pháp rà soát
- ❑ Danh mục cần rà soát
- ❑ Lợi ích của rà soát

Đảm bảo chất lượng phần mềm

27

## Quá trình rà soát

- ❑ Quá trình rà soát là cuộc họp mà các cá nhân phát triển dự án, quản lý dự án, và khách hàng tham gia xem xét, đánh giá, phê duyệt sản phẩm được phát triển ở mỗi giai đoạn.  
(IEEE 1990)

Đảm bảo chất lượng phần mềm

28

## Mục tiêu rà soát

- ❑ Mục tiêu trực tiếp
  - Tìm ra các khiếm khuyết và mục cần sửa chữa, thay đổi, hoàn thiện
  - Xác định rủi ro mới
  - Tìm ra những sai lệch so với mẫu chuẩn
  - Phê duyệt sản phẩm
- ❑ Mục tiêu gián tiếp
  - Tạo ra cuộc họp trao đổi kiến thức về phương pháp, công cụ và kỹ thuật phát triển

Đảm bảo chất lượng phần mềm

29

## Phương pháp rà soát

- ❑ Rà soát cá nhân (không chính thức, rà soát chéo)
- ❑ Rà soát kỹ thuật chính thức (Formal Technical Review)
- ❑ Rà soát ngang hàng (Peer review)
- ❑ Quan điểm của chuyên gia

Đảm bảo chất lượng phần mềm

30

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Là phương pháp rà soát duy nhất cần thiết để phê duyệt sản phẩm ở mỗi giai đoạn.
- ❑ FTRs phổ biến:
  - Rà soát kế hoạch phát triển
  - Rà soát đặc tả yêu cầu
  - Rà soát thiết kế sơ bộ, thiết kế chi tiết, thiết kế DB
  - Rà soát kế hoạch test, thủ tục test
  - Rà soát mô tả phiên bản
  - Rà soát hướng dẫn sử dụng, bảo trì PM
  - Rà soát kế hoạch cài đặt

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Vấn đề quan tâm
  - Thành viên tham gia
  - Chuẩn bị trước cho cuộc họp
  - Phiên rà soát chính thức
  - Hoạt động sau rà soát

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Thành viên tham gia rà soát:
  - **Người lãnh đạo rà soát**
    - Development department's manager
    - Chief software engineer
    - Leader of another project
    - Head of the SQA unit
  - **Nhóm thực hiện rà soát gồm 3 đến 5 người**
    - Senior members of the project team
    - Customer–user representatives
    - Software development consultants

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Chuẩn bị trước cuộc họp:
  - **Lãnh đạo rà soát:** bổ nhiệm các thành viên vào đội rà soát, lập lịch cho các phiên rà soát, phân phối tài liệu rà soát
  - **Nhóm rà soát:** rà soát tài liệu, viết ghi chú
  - **Nhóm phát triển:** chuẩn bị báo cáo sản phẩm cần rà soát

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Phiên rà soát chính thức:
  - Trình bày ngắn gọn tài liệu
  - Nhóm rà soát đưa ra ý kiến
  - Xác minh và thẩm định ý kiến
  - Đưa ra quyết định: chấp nhận, chấp nhận một phần, không chấp nhận

## Rà soát kỹ thuật chính thức(FTR)

- ❑ Hoạt động sau rà soát:
  - Báo cáo rà soát
    - Tóm tắt thảo luận rà soát
    - Quyết định của cuộc họp
    - Danh sách các vấn đề cần chỉnh sửa
    - Tên các thành viên theo dõi quá trình chỉnh sửa
  - Theo dõi



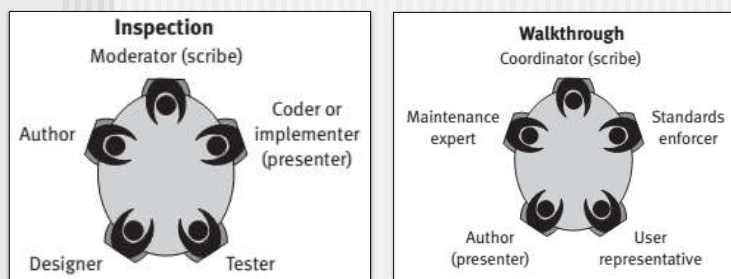
## Phương châm FTR

- ❑ Lập **checklist** cho từng sản phẩm cần rà soát
- ❑ **Đào tạo** cho các thành viên rà soát
- ❑ **Phân tích và soát trước** để cải tiến phương pháp rà soát
- ❑ **Lập lịch rà soát trong kế hoạch dự án** và cấp phát nguồn lực cho rà soát
- ❑ **Giới hạn số người tham gia** rà soát
- ❑ Phiên rà soát **không quá 2 giờ**
- ❑ **Hạn chế tranh luận, bác bỏ**
- ❑ Trình bày rõ ràng vấn đề, **không gượng ép giải quyết**
- ❑ Ghi chú lên bảng tường

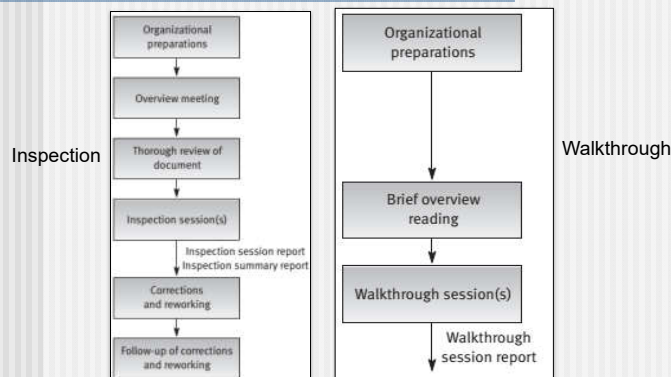
## Rà soát ngang hàng(peer review)

- ❑ Gồm 2 phương pháp phổ biến:
  - Inspection
  - Walkthrough

## Thành phần tham gia Peer Review



## Tiến trình Peer Review



## Rà soát ngang hàng(peer review)

- ❑ Các phần cần đưa vào:
  - Các phần có logic phức tạp
  - Phần quan trọng mà các khiếm khuyết của nó gây thiệt hại nghiêm trọng cho hệ thống
  - Các phần liên quan đến môi trường mới
  - Các phần được thực hiện bởi thành viên mới hoặc thiếu kinh nghiệm

## Rà soát ngang hàng(peer review)

- ❑ Các phần bỏ qua:
  - Các phần đơn giản
  - Phần thuộc loại đã được rà soát nhiều lần ở dự án tương tự
  - Các phần nếu có lỗi xảy ra không ảnh hưởng nhiều
  - Thiết kế và code được tái sử dụng

## So sánh các phương pháp rà soát

Thuộc tính	FTR	Inspection	Walkthrough
Mục tiêu trực tiếp	Tìm lỗi Xác định rủi ro mới Kiểm duyệt sản phẩm	Tìm lỗi	Tìm lỗi Tìm sự sai lệch so với chuẩn
Mục tiêu gián tiếp	Trao đổi kiến thức	Trao đổi kiến thức Hỗ trợ hoạt động sửa lỗi	Trao đổi kiến thức
Thành viên tham gia	Đội ngũ có vị trí cao	Ngang hàng	Ngang hàng
Sự tham gia của người có chuyên môn	No	Yes	Yes
Hợp sơ bộ	No	Yes	Yes
Chuẩn bị cho rà soát	Kỹ lưỡng	Kỹ lưỡng	Ngắn gọn
Phiên rà soát	Yes	Yes	Yes
Theo dõi chỉnh sửa	Yes	Yes	No

Đảm bảo chất lượng phần mềm

43

## Quan điểm của chuyên gia

- ❑ Khi nào cần ý kiến của chuyên gia bên ngoài?
  - Không đủ người có chuyên môn trong lĩnh vực chuyên ngành của dự án
  - Thiếu tạm thời người có chuyên môn cho nhóm rà soát
  - Sự bất đồng ý kiến giữa thành viên tham gia rà soát
  - Trong các tổ chức nhỏ, số lượng các ứng cử viên phù hợp cho nhóm rà soát không đủ.

Đảm bảo chất lượng phần mềm

44

## Danh mục cần rà soát

- ❑ Danh mục rà soát kế hoạch
- ❑ Danh mục rà soát đặc tả yêu cầu
- ❑ Danh mục rà soát kiến trúc
- ❑ Danh mục rà soát thiết kế mức cao(high-level)
- ❑ Danh mục rà soát thiết kế chi tiết
- ❑ Danh mục rà soát code
- ❑ Danh mục rà soát kiểm thử
- ❑ Danh mục rà soát bảo trì PM

Đảm bảo chất lượng phần mềm

45

## Danh mục rà soát kế hoạch

- ❑ Rà soát kế hoạch PM
  - Phạm vi PM đã được xác định chính xác
  - Nguồn lực có tương xứng với phạm vi PM, có sẵn, giá cả phù hợp?
  - Lịch biểu xác định cụ thể chưa, trình tự các công việc có logic, phù hợp với nguồn lực không?
  - Tổ chức nhân sự hợp lý chưa?
  - Rủi ro các thành phần đã xác định, đánh giá đầy đủ chưa?
  - Ngân sách, thời hạn dự kiến có khả thi?

Đảm bảo chất lượng phần mềm

46

## Danh mục rà soát yêu cầu

- ❑ Tổ chức và tính hoàn thiện
  - Tất cả các yêu cầu nhất quán và phù hợp?
  - Các yêu cầu cung cấp một cơ sở đầy đủ cho các thiết kế?
  - Xếp độ ưu tiên cho mỗi yêu cầu?
  - Các phần cứng, phần mềm, và các giao diện được xác định?
  - Đặc tả bao gồm đầy đủ yêu cầu KH và hệ thống?

Đảm bảo chất lượng phần mềm

47

## Danh mục rà soát yêu cầu

- ❑ Tính đúng đắn
  - Các yêu cầu mâu thuẫn hay trùng lặp?
  - Mỗi yêu cầu được viết rõ ràng, súc tích?
  - Mỗi yêu cầu nằm trong phạm vi dự án?
  - Mỗi yêu cầu không bị lỗi về nội dung và ngữ pháp?
  - Yêu cầu thiếu thông tin cần thiết?
  - Tất cả các yêu cầu thực hiện với các ràng buộc, phù hợp với nguồn lực, thời gian

Đảm bảo chất lượng phần mềm

48



## Danh mục rà soát yêu cầu

- ❑ **Đặc tính chất lượng**
  - Yêu cầu về hiệu năng, bảo mật đã được chỉ định rõ ràng chưa?
- ❑ **Mỗi yêu cầu có thể lần vết được ở mức hệ thống**
- ❑ **Các yêu cầu chức năng quan trọng và thời gian thực hiện chúng đã được xác định**

## Danh mục rà soát kiến trúc

- ❑ **Cấu trúc**
  - Kiến trúc có cho phép thực hiện tất cả các yêu cầu không?
  - Kiến trúc có được phân tách cấu trúc không?
  - Kiến trúc cung cấp cơ sở đầy đủ cho thiết kế?
  - Kiến trúc khả thi?
  - Khả năng bảo trì được xem xét chưa?
- ❑ **Tính đúng đắn**
  - Kiến trúc tránh dư thừa không cần thiết
  - Yêu cầu độ tin cậy, hiệu năng, bảo mật được giải quyết?
  - Cấu trúc dữ liệu đã được xác định chưa?

## Danh mục rà soát kiến trúc

- ❑ **Chuẩn và khả năng lần vết**
  - Kiến trúc có đúng chuẩn?
  - Các phần của kiến trúc có thể lần vết về các yêu cầu?
- ❑ **Tính logic**
  - Thiếu tính logic hay không?
- ❑ **Giao diện**
  - Tất cả các giao diện được xác định, rõ ràng?
  - Dữ liệu tối thiểu truyền qua giao diện?
- ❑ **Luồng dữ liệu, luồng điều khiển và giao diện của kiến trúc được biểu diễn rõ ràng?**

## Danh mục rà soát thiết kế mức cao(high-level)

- ❑ **Cấu trúc**
  - Thiết kế có thể phân tách được thành những phần nhỏ cho pha thiết kế chi tiết?
  - Modun hóa có phù hợp với chuẩn và hiệu quả?
- ❑ **Dữ liệu**
  - Dữ liệu có được truyền theo dạng tham số?
  - Cấu trúc dữ liệu và tên của các phần tử có ý nghĩa và phù hợp với qui ước đặt tên?

## Danh mục rà soát thiết kế mức cao(high-level)

- ❑ **Tính đúng đắn và hoàn thiện**
  - Thiết kế đáp ứng các yêu cầu?
  - Cơ sở hạ tầng được giải quyết?
  - Ưu tiên xử lý được giải quyết?
  - Thiết kế có tính khả thi và kinh tế?
  - Bộ nhớ cần thiết để lưu trữ modun, file, table đủ không?
  - Tính bảo mật được thiết kế chưa?
  - Thiết kế đã xem xét các ràng buộc, khả năng bảo trì?
  - Thiết kế hoàn thiện, chính xác, rõ ràng
- ❑ **Chuẩn và khả năng lần vết**
  - Các chuẩn thiết kế được đáp ứng?
  - Giao diện người dùng tuân theo chuẩn chung của dự án
  - Các thành phần của

## Danh mục rà soát thiết kế mức cao(high-level)

- ❑ **Giao diện**
  - Tất cả các giao diện rõ ràng
  - Dữ liệu tối thiểu được truyền qua mỗi giao diện?
- ❑ **Chuẩn và khả năng lần vết**
  - Các chuẩn thiết kế được đáp ứng?
  - Giao diện người dùng tuân theo chuẩn chung của dự án
  - Các thành phần của
- ❑ **Cơ chế xử lý lỗi được xác định chưa?**

## Danh mục rà soát thiết kế chi tiết

### □ Cấu trúc

- Mã giả hoặc dạng biểu diễn khác có thống nhất không?

### □ Dữ liệu

- Tất cả các dữ liệu đã được định nghĩa hoặc khởi tạo?
- Tất cả các dữ liệu đã được sử dụng?
- Kiểu dữ liệu và tên biến có phù hợp?

## Danh mục rà soát thiết kế chi tiết

### □ Tính đúng đắn và hoàn thiện

- Thiết kế chi tiết thực hiện chính xác, hoàn chỉnh?
- Các chức năng xác định rõ ràng, độc lập logic?
- Modun có độc lập chức năng?
- Thiết kế được xác minh?
- Tất cả giao diện người dùng được thiết kế hoàn chỉnh?

### □ Chuẩn và khả năng lần vết

- Thiết kế chi tiết tuân theo chuẩn?
- Giao thức gọi theo chuẩn của dự án?
- Các thành phần có thể lần vết về thiết kế mức cao và các yêu cầu

## Danh mục rà soát code

### □ Cấu trúc

- Code thực hiện đúng với thiết kế?
- Phù hợp với chuẩn code?
- Định dạng, cấu trúc, nhất quán?
- Có tồn tại mã chết, thủ tục không cần thiết, không được gọi?
- Có tồn tại đoạn code lặp lại?
- Có tồn tại module quá phức tạp cần cấu trúc lại?

### □ Tài liệu

- Code được viết rõ ràng, dễ bảo trì?
- Các chú thích nhất quán với code?

### □ Biến

- Các biến được khai báo, tên rõ ràng...?
- Các biến được gán giá trị, chuyển kiểu phù hợp?
- Có tồn tại các biến dư thừa?

## Danh mục rà soát code

### □ Toán tử

- Code có ngăn chặn lỗi làm tròn?
- Phép chia đã được kiểm tra?
- Phép cộng/trừ với các số có độ khác biệt lớn?

### □ Vòng lặp, nhánh

- Vòng lặp, nhánh xây dựng logic, đầy đủ, chính xác, và lồng nhau đúng cách?
- Các trường hợp đã bảo phủ đủ trong IF-ELSE, SWITCH-CASE?
- Điều kiện kết thúc vòng lặp rõ ràng
- Giá trị biến chạy được khởi tạo đúng chưa?
- Có tồn tại câu lệnh sai vị trí so với vòng lặp

## Danh mục rà soát code

### □ Lập trình phòng thử

- Con trỏ, mảng, chỉ số của mảng, file đã được kiểm tra chưa?
- Dữ liệu đầu vào đã được kiểm tra tính hợp lệ, hoàn thiện chưa?
- Tất cả các biến đầu ra đã được gán giá trị chưa?
- Bộ nhớ được cấp phát có được thu hồi không?

## Danh mục rà soát kiểm thử

### □ Rà soát kiểm thử

- Kế hoạch KT có phù hợp với kế hoạch dự án?
- Lịch trình có xác định rõ ràng, cụ thể?
- Nguồn lực, công cụ có sẵn chưa?
- Các ca kiểm thử được xác định đầy đủ và sắp xếp độ ưu tiên chưa?
- Điều kiện kiểm thử được xác định?
- Phương pháp kiểm thử được đặc tả chưa?

## Danh mục rà soát bảo trì PM

### □ Rà soát bảo trì PM

- Các yêu cầu thay đổi, thêm/bớt đã được lập tài liệu chưa?
- Xem xét các hiệu ứng xảy ra khi yêu cầu thay đổi?
- Các thay đổi đã được báo cáo với các bên liên quan chưa?

## Lợi ích của rà soát

- Cải tiến quá trình phát triển PM, giảm thời gian phát triển
- Tìm lỗi hiệu quả, giảm mức độ lỗi → tăng chất lượng sản phẩm
- Chi phí test được giảm từ 50% -> 80%
- Giảm chi phí bảo trì 28 lần
- Loại bỏ khoảng 80%-95% các lỗi ở mỗi bước