

CÔNG NGHỆ PHẦN MỀM

SOFTWARE ENGINEERING

TS. Võ Đức Hoàng

Khoa Công nghệ thông tin

Trường Đại học Bách khoa - Đại học Đà Nẵng

■ ***Giáo trình chính:***

- Andrew Troelsen, *Pro C# 5.0 and The .NET 4.5 Framework*, Apress, 2012.

■ ***Tài liệu tham khảo:***

- Simon Kendal, *Object Oriented Programming using C#*, BookBoon, 2012.
- Microsoft MSDN, *C# Programming Guide for Visual Studio 2013*.
- Joe Mayo, *LINQ Programming*, McGraw-Hill Education, 2009.
- Andrew Clymer, *Pro Asynchronous Programming with .NET*, Apress, 2013...

Quản trị dự án (8)

- Tại sao quản trị dự án?
- Các hoạt động quản trị dự án.

Tại sao quản trị dự án?

- Quản trị dự án là cần thiết để thực hiện phần mềm
 - Đúng tiến độ
 - Giảm chi phí
 - Đạt được mục tiêu
- Quản trị dự án là rất quan trọng vì
 - Dự án phần mềm phức tạp
 - Sự thay đổi thường xuyên xuất hiện trong quá trình phát triển
 - Cần đảm bảo các ràng buộc
 - * Thời gian
 - * Chi phí
 - * Nguồn tài nguyên

Các hoạt động quản trị dự án



- Lập kế hoạch
 - Xác định các hoạt động cần thực hiện
- Lập lịch
 - Lập lịch cho các hoạt động, đảm bảo đúng tiến độ
- Tổ chức
 - Chọn lựa, đánh giá, phân công công việc cho các thành viên
- Định giá
 - Ước lượng chi phí
 - Nhân lực
 - Nguồn tài nguyên cần thiết

Các hoạt động quản trị dự án



■ Lãnh đạo

- Đưa ra các quyết định
- Đảm bảo sự hợp tác giữa các thành viên trong nhóm

■ Giám sát

- Kiểm tra tiến độ
- Giám sát chi phí/nhân lực

■ Hiệu chỉnh

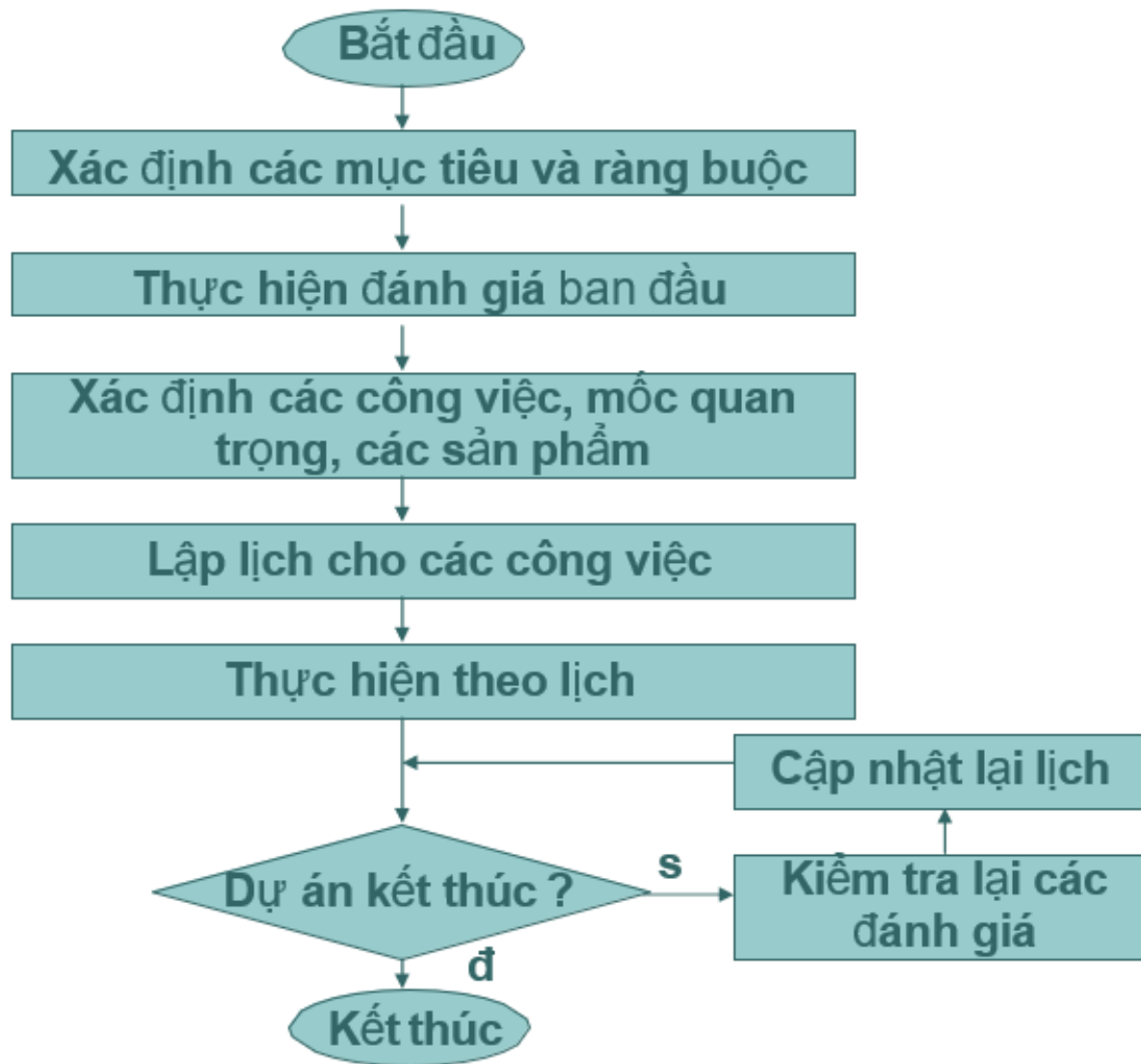
- Có các biện pháp hiệu chỉnh cần thiết nếu dự án bị chậm trễ

■ Lập báo cáo

- Viết các báo cáo, trình bày

- Quản lý hiệu quả dự án phụ thuộc vào kế hoạch
- Được thực hiện trong suốt quá trình thực hiện dự án
- Lập kế hoạch bao gồm xác định:
 - Các mục tiêu
 - Các ràng buộc
 - Các công việc cần thực hiện để đạt mục tiêu
 - Các mốc quan trọng (milestones)
 - Các sản phẩm tạo ra

Lập kế hoạch



■ Xác định mục tiêu

- Mục tiêu chung của dự án
- Các chức năng cơ bản mà phần mềm phải đáp ứng
- Yêu cầu về chất lượng

■ Các ràng buộc

- Ngày giao sản phẩm
- Nhân sự
- Ngân sách cho phép
- Thiết bị, phần cứng
- Phương thức giao tiếp với khách hàng
- ...

- Đánh giá ban đầu các tham số của dự án
 - Cấu trúc
 - Kích thước
 - Chi phí
 - Phân tích các chức năng của phần mềm
 - Nhân công
 - Nhân lực yêu cầu

- Các mốc quan trọng (milestones)
 - Các bước hoàn thành quan trọng của dự án
 - Ví dụ: thẩm định đặc tả yêu cầu, thẩm định thiết kế
 - Các mốc quan trọng cho phép giám sát được tiến độ
- Xác định các sản phẩm (deliverables) trong các bước bàn giao cho khách hàng
 - Đặc tả yêu cầu
 - Nguyên mẫu
 - Thiết kế giao diện người dùng
 - ...

- Dự án cần phải chia thành các công việc (task/activity)
- Các công việc không nên quá nhỏ
 - Mỗi công việc nên kéo dài khoảng 2 tuần
- Mỗi công việc tiếp tục được chia thành các công việc con dễ dàng xử lý
- Một công việc con dễ dàng xử lý
 - Có kết quả dễ dàng đánh giá
 - Dễ thực hiện
 - Dễ đánh giá thời gian thực hiện
 - Dễ đánh giá nhân công, tài nguyên cần thiết

- Chia công việc
 - Một cách đơn giản để xác định và chia công việc là tạo WBS (Work Breakdown Structure)
 - tương tự như một mục lục
 - Ví dụ
 - Khởi động dự án
 - * Lập kế hoạch dự án
 - Phân tích yêu cầu
 - * Thu thập yêu cầu
 - * Mô hình hóa yêu cầu sử dụng UML
 - Thiết kế
 - * Xây dựng các biểu đồ lớp
 - * Xây dựng các biểu đồ tuần tự
 - * Xây dựng các biểu đồ gói
 - Mã hóa
 - Kiểm thử

■ Cần chứa các mục (1)

- Giới thiệu
 - mô tả mục tiêu
 - ràng buộc
- Tổ chức
 - các thành viên của nhóm
 - vai trò của các thành viên
- Phân tích rủi ro
 - dự báo các rủi ro có thể
 - đề xuất các giải pháp hạn chế rủi ro
- Nguồn tài nguyên cần thiết
 - phần cứng
 - phần mềm

■ Cần chứa các mục (2)

- Chia công việc

- chia dự án thành các công việc
- xác định các mốc quan trọng
- xác định nội dung các sản phẩm giao hàng

- Lịch

- mô tả ràng buộc các công việc và thời gian để đạt được các mốc quan trọng
- gán công việc cho các thành viên

- Giám sát

- mô tả các báo cáo được tạo ra khi nào và như thế nào
- mô tả cơ chế sử dụng để thực hiện thẩm định các công việc đã hoàn thành

- Lập lịch bao gồm các công việc
 - xác định ngày quan trọng
 - ngày bắt đầu, ngày kết thúc
 - xác định các giai đoạn quan trọng
 - liệt kê các công việc trong thứ tự thực hiện
 - chỉ ra quan hệ giữa các công việc
 - đánh giá nguồn tài nguyên cần thiết để hoàn thành mỗi công việc
 - nhân lực, thời gian, ngân sách

- Liệt kê các công việc trong thứ tự thực hiện
- chỉ ra sự phụ thuộc giữa các công việc
 - các công việc nào có thể tiến hành đồng thời
 - các công việc nào chỉ thực hiện khi công việc khác kết thúc
- giảm tối thiểu các phụ thuộc
 - hạn chế sự chậm trễ
- thời gian thực hiện dự án phụ thuộc con đường dài nhất trong đồ thị công việc
 - sơ đồ PERT

- Sử dụng bảng để biểu diễn lịch của dự án
 - Bảng các giai đoạn quan trọng
 - Bảng các công việc
 - Bảng phân công

- Bảng các giai đoạn quan trọng
- các giai đoạn quan trọng và ngày có thể đạt được

Ngày	Giai đoạn quan trọng
August 26	Project Kickoff (with client)
October 16	Analysis Review
October 26	System Design Review
November 7	Internal Object Design Review
November 20	Project Review (with client)
Nov 26	Internal project review
Dec 11	Acceptance test (with client)

- Bảng các công việc
- các công việc và ngày bắt đầu/ngày kết thúc

Ngày

Jul 17-Aug 23

Aug 26 - Sep 24

Sep 11-Oct 8

Oct 9 - Oct 26

Oct 28-Nov 7

Nov 8 - Nov 20

Nov 22 - Dec 4

Dec 4 - Dec 10

Dec 11- Dec 18

Công việc

Preplanning Phase

Project Planning

Requirements Analysis

System Design

Object Design

Implementation & Unit Testing

System Integration Testing

System Testing

Post-Mortem Phase

- Bảng phân công
- ai làm gì và thời gian bao lâu

Công việc	Phân công	Thời gian (người/ngày)	Phụ thuộc
T1	Jane	8	
T2	Anne (75%)	15	
T3	Jane (80%)	15	T1 (M1)
T4	Fred	10	
T5	Mary	10	T2, T4 (M2)
T6	Anne	5	T1, T2 (M3)
T7	Jim	20	T1 (M1)
T8	Fred	25	T4 (M5)
T9	Jane	15	T3, T6 (M4)
T10	Anne	15	T5, T7 (M7)
T11	Fred	7	T9 (M6)
T12 ¹⁹	Fred (50%)	10	T11 (M8)

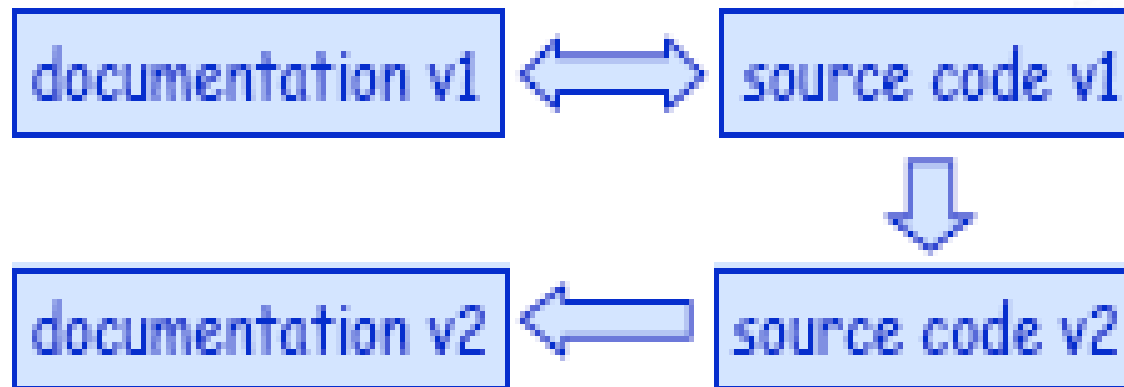
- Có thể sử dụng các sơ đồ để xây dựng, phân tích các lịch phức tạp
 - Sơ đồ Gantt
 - biểu diễn quan hệ thời gian giữa con người và công việc
 - Sơ đồ PERT
 - biểu diễn phụ thuộc giữa các công việc

■ Tài liệu là cần thiết cho chương trình

- để sử dụng chương trình
 - cần mô tả đầy đủ về chương trình
 - * mục đích, môi trường, thuật toán, vào/ra, thời gian thực thi...
- để tin tưởng chương trình
 - báo cáo kết quả kiểm thử
 - * kiểm thử các chức năng thực hiện tốt
 - * kiểm thử các tình huống không mong đợi
- để chỉnh sửa chương trình
 - mô tả đầy đủ chương trình
 - cấu trúc bên trong
 - mô tả vết chỉnh sửa

- Những người sử dụng khác nhau yêu cầu các loại tài liệu khác nhau
 - người sử dụng
 - tài liệu hướng dẫn sử dụng
 - người phát triển
 - tài liệu phát triển
 - chú thích
 - người thiết kế
 - mô hình thiết kế
 - người quản lý
 - kết quả kiểm thử

- Cần duy trì sự gắn kết giữa mã nguồn và tài liệu



■ Vấn đề

- cần duy trì sự gắn kết giữa mã nguồn và tài liệu trong các tệp khác nhau

■ Giải pháp

- xây dựng tài liệu tự động (auto-documentation)
 - Javadoc, CcDoc, CcpDoc, AutoDoc, DocClass...
- sinh mã tự động từ mô hình thiết kế
- sinh mô hình thiết kế từ mã nguồn
 - Rational Rose, Jude, Poseidon, ArgoUML...

- Cấu hình phần mềm bao gồm
 - các thành phần phần mềm xác định tính chất cơ bản của phần mềm
 - một thành phần có thể
 - mã nguồn, tệp dữ liệu, đặc tả yêu cầu, tài liệu thiết kế, cấu hình phần cứng...

- Quản lý cấu hình là lĩnh vực của quản trị dự án nhằm
 - định nghĩa
 - xác định
 - quản lý
 - kiểm tra
 - cấu hình trong suốt quá trình phát triển phần mềm
- Định nghĩa IEEE (Standard 1042)

“Software configuration management (SCM) is the discipline of managing and controlling change in the evolution of software systems”

- SCM để hỗ trợ người quản lý
 - giám sát các thay đổi trong quá trình phát triển
 - gồm các hoạt động
 - xây dựng các thử cần thực hiện khi có sự thay đổi
 - ghi nhận các thành phần và yêu cầu thay đổi
 - đo lường chi phí và công sức thực hiện thay đổi
 - ...
- SCM để hỗ trợ người phát triển
 - cung cấp chức năng và công cụ hỗ trợ người phát triển thực hiện các thay đổi
 - gồm các hoạt động
 - quản lý các chức năng khác nhau của phần mềm
 - xây dựng lại cấu hình trước đó
 - ghi nhận vết thay đổi của của phần mềm
 - ...

■ Gồm các hoạt động (1)

- định nghĩa các thành phần của cấu hình
 - các loại tài liệu cần quản lý
 - * Đặc tả yêu cầu, tài liệu thiết kế, mã nguồn, báo cáo kiểm thử...
- định nghĩa **chính sách** quản lý thay đổi và quản lý phiên bản
 - mục đích của chính sách thay đổi nhằm đảm bảo mỗi phiên bản đáp ứng tiêu chuẩn đặt ra
 - ví dụ
 - * “không phân phối sản phẩm cho khách hàng nếu chưa thực hiện bước kiểm thử beta với ít nhất 1000 người sử dụng bên ngoài”

- Gồm các hoạt động (2)
 - định nghĩa vai trò và trách nhiệm của các thành viên trong các hoạt động SCM
 - người quản lý, người phát triển...
 - định nghĩa CSDL sử dụng để ghi thông tin về cấu hình
 - định nghĩa các công cụ sử dụng hỗ trợ SCM
 - Chọn lựa chuẩn để sử dụng
 - Ví dụ
 - * IEEE 828-1990: Software Configuration Management Plans
 - * IEEE 1042: Guide to Software Configuration Management

- Phần mềm thường xuyên thay đổi do yêu cầu của
 - người sử dụng
 - người phát triển
 - thị trường

- Quản lý thay đổi là ghi nhận tất cả các sự thay đổi và bảo đảm rằng chúng được thực hiện với chi phí thấp nhất

- Thuật ngữ
 - promotion
 - một phiên bản được chuyển giao cho các người phát triển
 - release
 - một phiên bản được chuyển giao cho người sử dụng (ngoài nhóm phát triển)
- Đặt tên các phiên bản
 - rõ ràng, không nhập nhằng
 - phương pháp đơn giản thường được sử dụng
 - đánh số

- Biên dịch và kết hợp tất cả các thành phần của một cấu hình thành một hệ thống thực thi được
- Các cách kết hợp khác nhau các thành phần có thể tạo nên các hệ thống khác nhau
- Nên sử dụng các công cụ hỗ trợ
 - Ví dụ: Makefile

- Các vấn đề cần lưu ý khi xây dựng hệ thống:
 - Tất cả các thành phần cần thiết đều được sử dụng (liên kết)?
 - Phiên bản thích hợp của mỗi thành phần được sử dụng?
 - Tất cả các tệp dữ liệu đã sẵn sàng?
 - Hệ thống được xây dựng cho nền (platform) đúng đắn?
 - hệ điều hành, cấu hình phần cứng
 - Phiên bản của trình biên dịch và các công cụ sử dụng là đúng đắn?

- SCM được hỗ trợ bởi các công cụ
- Có các loại công cụ
 - các công cụ độc lập
 - các công cụ tích hợp vào trong các môi trường phát triển

- Hoạt động hỗ trợ
 - Đặt tên các phiên bản
 - tự đặt tên các phiên bản mới
 - Ghi lại lịch sử (vết) thay đổi
 - Phát triển cộng tác
 - nhiều người có thể thay đổi đồng thời một phiên bản
 - Ghi nhận các phiên bản: 2 khả năng
 - Ghi nhận toàn bộ phiên bản
 - Chỉ ghi nhận sự khác nhau giữa các phiên bản

- Tổ chức dự án là rất quan trọng
 - Yếu tố chính quyết định cho sự thành công
- Bao gồm các hoạt động
 - Chọn nhân sự thích hợp
 - Chọn cấu trúc của nhóm
 - Chọn kích thước của nhóm
 - Xác định vai trò của các thành viên trong nhóm
 - Quản lý giao tiếp giữa các thành viên trong nhóm

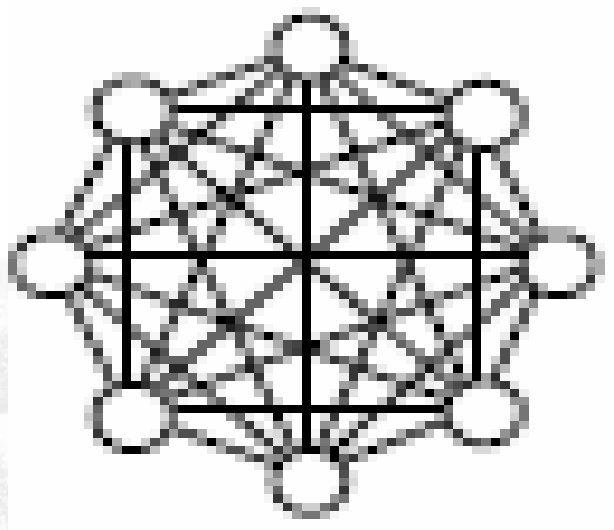
- Các yếu tố cần xem xét khi chọn nhân sự
 - Kinh nghiệm
 - hiểu biết lĩnh vực ứng dụng
 - kinh nghiệm với môi trường phát triển
 - hiểu biết về ngôn ngữ lập trình
 - Đào tạo
 - Khả năng
 - khả năng giao tiếp
 - khả năng thích ứng, khả năng học
 - Thái độ
 - Tính cách

Tổ chức dự án - Chọn cấu trúc của nhóm



- Nhóm không hình thức (egoless team)
- Nhóm chief-programmer
- Nhóm phân cấp

- Nhóm phi hình thức (egoless team)
 - các thành viên của nhóm có vai trò như nhau
 - nhóm nhỏ
 - các thành viên đều có kinh nghiệm và năng lực
 - dự án khó



■ Nhóm chief-programmer

- Gồm có

- Trưởng nhóm (chief-programmer): thực hiện phân tích, thiết kế, mã hóa, kiểm thử
- Trợ lý: hỗ trợ trưởng nhóm phát triển, kiểm thử
- Thư ký: quản lý thông tin
- Các chuyên gia hỗ trợ
 - * quản lý, lập tài liệu, lập trình, kiểm thử...

- Phụ thuộc chủ yếu vào trưởng nhóm

- Trưởng nhóm phải có năng lực

- Nhóm phân cấp
 - Dự án lớn được chia thành nhiều dự án nhỏ
 - Mỗi dự án nhỏ được hiện bởi một nhóm
 - Mỗi nhóm có một trưởng nhóm
 - Mỗi thành viên cấp dưới phải báo cáo công việc với người quản lý trực tiếp
 - Mỗi thành viên phải được đào tạo kỹ năng để thực hiện vai trò của mình

Tổ chức dự án - Chọn kích thước của nhóm



- Kích thước nhóm nên tương đối nhỏ: dưới 8 người
 - giảm thời gian giao tiếp
 - dễ dàng làm việc cùng nhau
- Không nên quá nhỏ
 - nhóm bảo đảm tiếp tục làm việc, nếu có thành viên ra
- đi
- Đối với một dự án, số người trong nhóm có thể thay đổi
- Khi một dự án chậm trễ, thêm người vào dự án không bao giờ giải quyết được vấn đề

“Adding more programmers to a late project makes it later” (Brooks’ Law - The Mythical Man-Month)

■ Trưởng dự án

- chịu trách nhiệm một dự án
- bảo đảm nhóm có đầy đủ thông tin và nguồn tài nguyên cần thiết
- phân công công việc cho các thành viên
- kiểm tra thời hạn các công việc
- giao tiếp với khách hàng

- Giao tiếp tốt cho phép nhóm hoạt động tốt
- Thông tin cần trao đổi về
 - tiến độ công việc
 - các thay đổi
 - các khó khăn
 - ...
- Giao tiếp giữa các thành viên phụ thuộc vào cấu trúc nhóm
 - nhóm phi hình thức: giao tiếp trực tiếp giữa các thành viên
 - nhóm phân cấp: giao tiếp thông qua người quản lý

- Các đặc điểm trong giao tiếp nhóm (1)
 - các thành viên có vị trí cao thường áp đặt các cuộc trao đổi
 - nhóm vừa có nam và nữ thường giao tiếp tốt hơn
 - giao tiếp phải qua một người điều phối trung tâm thường không hiệu quả
 - tất cả các thành viên nên có tham gia vào các quyết định ảnh hưởng toàn bộ nhóm

■ Các đặc điểm trong giao tiếp nhóm (2)

- Tính cách của các thành viên

- quá nhiều thành viên có cùng tính cách cũng có thể không tốt
 - * hướng công việc: mỗi người đều muốn thực hiện công việc riêng
 - * hướng cá nhân: mỗi người đều muốn làm ông chủ
 - * hướng tương tác: nhiều hợp hành mà ít thực hiện cụ thể
- một nhóm nên cân bằng giữa các tính cách

- Rủi ro (risk) là khả năng một tính huống xấu xảy ra
- **Quản lý rủi ro** (risk management) liên quan đến
 - xác định các rủi ro ảnh hưởng đến dự án
 - lập kế hoạch hạn chế sự ảnh hưởng của rủi ro
- Các loại rủi ro
 - rủi ro của dự án (project risks) ảnh hưởng đến tiến độ và nguồn tài nguyên
 - rủi ro của sản phẩm (product risks) ảnh hưởng đến chất lượng phần mềm
 - rủi ro của doanh nghiệp (enterprise risks) ảnh hưởng đến doanh nghiệp sẽ sử dụng phần mềm

- Rủi ro (risk) là khả năng một tính huống xấu xảy ra
- **Quản lý rủi ro** (risk management) liên quan đến
 - xác định các rủi ro ảnh hưởng đến dự án
 - lập kế hoạch hạn chế sự ảnh hưởng của rủi ro
- Các loại rủi ro
 - rủi ro của dự án (project risks) ảnh hưởng đến tiến độ và nguồn tài nguyên
 - rủi ro của sản phẩm (product risks) ảnh hưởng đến chất lượng phần mềm
 - rủi ro của doanh nghiệp (enterprise risks) ảnh hưởng đến doanh nghiệp sẽ sử dụng phần mềm

- Các hoạt động quản lý rủi ro
 - **Xác định** các rủi ro
 - **Phân tích** các rủi ro
 - **Lập kế hoạch** các rủi ro
 - **Giám sát** các rủi ro
 - **Xử lý** các rủi ro

- Rủi ro về thương mại
 - Đối thủ cạnh tranh có chiếm lĩnh thị trường trước?
 - Có cần cho ra đời phiên bản nhỏ để chiếm thị trường?
- Rủi ro về tài chính
 - Có đủ năng lực về tài chính để thực hiện dự án đúng tiến độ?
- Rủi ro về kỹ thuật
 - Công nghệ hiện tại có cho phép?
- Rủi ro về con người
 - Nhóm làm việc có đủ kinh nghiệm và năng lực?

- Rủi ro về thương mại
 - Đối thủ cạnh tranh có chiếm lĩnh thị trường trước?
 - Có cần cho ra đời phiên bản nhỏ để chiếm thị trường?
- Rủi ro về tài chính
 - Có đủ năng lực về tài chính để thực hiện dự án đúng tiến độ?
- Rủi ro về kỹ thuật
 - Công nghệ hiện tại có cho phép?
- Rủi ro về con người
 - Nhóm làm việc có đủ kinh nghiệm và năng lực?

- Kế hoạch giảm rủi ro cho mỗi rủi ro gồm
 - tầm quan trọng đối với khách hàng
 - tầm quan trọng đối với người phát triển
 - chiến lược quản lý rủi ro và ảnh hưởng về kinh tế
 - phương tiện kiểm tra rủi ro đã bị xóa hoặc đã giảm
 - các kịch bản bị ảnh hưởng bởi rủi ro

■ Các chiến lược

- Chiến lược tránh rủi ro
 - giảm xác suất rủi ro xảy ra
- Chiến lược giảm rủi ro
 - giảm ảnh hưởng của rủi ro đối với dự án hoặc sản phẩm khi nó xảy ra
- Kế hoạch khẩn cấp
 - xử lý ngay rủi ro khi xảy ra

- Giám sát các rủi ro
 - Đánh giá thường xuyên mỗi rủi ro
 - để xác định xác suất xảy ra của nó
 - để đánh giá các hậu quả của nó có thay đổi
 - Mỗi rủi ro chính cần phải được thảo luận khi có các cuộc họp về tiến độ dự án
- Xử lý các rủi ro
 - Phương án xử lý khi rủi ro xảy ra

