



TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

# NHẬP MÔN CNPM

## Nội dung / Chương 12-1: Giới thiệu về QLDA

Thông tin GV

# Nội dung chương

Phần 1. ABC

Phần 2. ABC

Phần 3. ABC

Phần 4. ABC

Phần 5. ABC

Phần 6. ABC

# Bốn yếu tố

- **Con người**— yếu tố quan trọng nhất của dự án
- **Sản phẩm** — phần mềm được xây dựng
- **Quy trình** — tập hợp các hoạt động chương trình khung và nhiệm vụ của phần mềm để thực hiện công việc
- **Dự án** — tất cả các công việc để sản phẩm trở thành hiện thực

# Các thành phần liên quan

- **Quản lý cấp cao:** người xác định vấn đề nghiệp vụ, thường có ảnh hưởng lớn tới dự án.
- **Quản lý dự án(kỹ thuật):** người lên kế hoạch, động viên, tổ chức và kiểm soát những người thực hiện phần mềm.
- **Người thực hiện:** có kỹ năng kỹ thuật cần thiết để thiết kế sản phẩm hoặc ứng dụng.
- **Khách hàng:** người đưa ra các yêu cầu cho phần mềm được thiết kế.
- **Người sử dụng cuối cùng:** người tương tác với phần mềm khi nó được phát hành.

# Nhóm phần mềm

**Làm sao để dẫn dắt?**

**Làm sao để hợp tác?**

**Làm sao để tổ chức?**



**Làm thế nào để thúc đẩy?**

**Làm sao để tạo ra những ý tưởng tốt?**

# Team Leader

- Mô hình MOI (Motivation- Organization- Ideas or innovation)
  - **Động viên.** Khả năng khích lệ (bằng cách kéo hoặc đẩy) nhân viên kỹ thuật để tạo ra sản phẩm với khả năng tốt nhất
  - **Tổ chức.** Khả năng định hình quá trình hiện có (hoặc phát minh ra cái mới) để có thể từ ban đầu tạo ra sản phẩm cuối cùng
  - **Ý tưởng và cải tiến.** Khả năng khuyến khích mọi người tạo ra và cảm thấy sáng tạo ngay cả khi họ làm việc trong giới hạn đã được thiết lập của sản phẩm hay ứng dụng

# Nhóm phần mềm

- Các yếu tố sau đây phải được xem xét khi lựa chọn một cơ cấu nhóm dự án phần mềm...
  - Mức độ khó khăn các vấn đề cần giải quyết
  - Kích thước chương trình kết quả trên cơ sở LOC hoặc số lượng hàm.
  - Thời gian làm việc cùng nhau
  - Cấp độ các vấn đề có thể module hóa
  - Yêu cầu chất lượng và độ tin cậy của hệ thống được xây dựng
  - Mức độ nghiêm khắc của ngày chuyển giao
  - Mức độ xã hội (giao tiếp) của dự án

# Các Mô hình tổ chức

- **Mô hình đóng**—cấu trúc một nhóm với sự phân quyền truyền thống
- **Mô hình ngẫu hứng**—cấu trúc một nhóm lỏng lẻo và phụ thuộc vào sáng kiến cá nhân của mỗi thành viên
- **Mô hình mở**—cố gắng cấu trúc một nhóm mà cách thi hành kết hợp giữa mô hình chặt chẽ và mô hình ngẫu hứng
- **Mô hình đồng bộ**—dựa trên sự phân chia tự nhiên của một vấn đề và các thành viên trong nhóm sẽ làm việc trên các thành phần của vấn đề đó với ít sự giao tiếp lẫn nhau giữa các thành phần của vấn đề

*suggested by Constantine [Con93]*



# Tránh những nhóm “độc”

- Không khí làm việc điên cuồng, các thành viên nhóm lãng phí năng lượng và mất tập trung vào các mục tiêu của công việc cần được thực hiện
- Sự tắc trở gây ra bởi cá nhân, doanh nghiệp hay yếu tố công nghệ gây ra sự xích mích giữa các thành viên trong nhóm
- “phối hợp không ăn khớp” hoặc mô hình quy trình không phù hợp sẽ trở thành rào cản để hoàn thành
- Không xác định rõ ràng về vai trò dẫn đến thiếu trách nhiệm và kết quả là việc chỉ nói mà không làm
- Nhắc lại thất bại nhiều lần dẫn tới mất tự tin và hạ tinh thần.

# Nhóm Agile (nhanh nhẹn)

- Các thành viên trong nhóm có niềm tin vào nhau
- Phân phối kỹ năng phù hợp với vấn đề cần giải quyết
- Loại bỏ đại diện không cần thiết, nếu sự gắn kết được duy trì
- Nhóm “tự tổ chức”
  - Cấu trúc nhóm có khả năng thích nghi
  - Sử dụng các yếu tố ngẫu nhiên, mở và mô hình đồng bộ
  - Quyền tự chủ lớn

# Phối hợp và giao tiếp nhóm

- **Chính thức, cách tiếp cận cá nhân:** bao gồm các tài liệu kỹ thuật phần mềm và các sản phẩm công việc (bao gồm cả mã nguồn), bản ghi nhớ kỹ thuật, sự kiên quan trọng của dự án, lịch trình và các công cụ kiểm soát dự án.(chương 23), yêu cầu thay đổi và các tài liệu liên quan, báo cáo theo dõi lỗi, và dữ liệu kho lưu trữ (xem Chương 26)
- **Chính thức, thủ tục giao tiếp cá nhân:** tập trung vào các hoạt động đảm bảo chất lượng (Chương 25) áp dụng cho các sản phẩm công việc kỹ thuật phần mềm. Chúng bao gồm các cuộc họp đánh giá tình trạng và kiểm tra thiết kế, code.
- **Không chính thức, thủ tục giao tiếp cá nhân:** bao gồm các cuộc họp nhóm để phổ biến thông tin và giải quyết vấn đề và sắp xếp các yêu cầu của thành viên
- **Giao tiếp điện tử:** thông qua thư điện tử và họp qua truyền hình.
- **Mạng lưới giao tiếp cá nhân:** bao gồm các cuộc tranh luận không chính thức với thành viên trong nhóm hoặc người ngoài nhóm để trợ trong công việc

# Phạm vi sản phẩm

- Phạm vi
  - **Hoàn cảnh**: Làm thế nào để phần mềm được xây dựng phù hợp với hệ thống, sản phẩm hoặc hoàn cảnh kinh doanh lớn hơn, và các ràng buộc nào của hoàn cảnh
  - **Mục tiêu Thông tin**: khách hàng thấy được sản phẩm đầu ra của phần mềm là gì? Dữ liệu yêu cầu của đầu vào là gì?
  - **Chức năng và hiệu suất**: chức năng nào của phần mềm thực hiện biến đổi dữ liệu đầu vào thành đầu ra? Tất cả trường hợp riêng đặc biệt đều được giải quyết?
- Phạm vi dự án phần mềm phải rõ ràng và dễ hiểu tại mỗi mức quản lý và chuyên môn

# Phân tách Vấn đề

- Đôi khi được gọi là **sự phân hoạch** hoặc **giải thích vấn đề**
- Khi phạm vi được xác định...
  - Nó được phân rã thành các chức năng cấu thành
  - Nó được phân rã thành đối tượng dữ liệu người dùng có thể thấy
  - Nó được chia ra thành một tập các lớp vấn đề
- Quá trình phân hoạch tiếp tục cho đến khi tất cả các chức năng hoặc các lớp vấn đề đã được xác định

# Quy trình

- Khi khuôn khổ quy trình được xác lập
  - Xem xét các đặc điểm của dự án
  - Xác định mức độ nghiêm khắc của yêu cầu
  - Xác định nhiệm vụ đặt ra cho mỗi hoạt động kỹ thuật phần mềm
    - **Nhiệm vụ =**
      - Tác vụ kỹ thuật phần mềm
      - Sản phẩm công việc
      - Sự đảm bảo chất lượng
      - Mốc đạt được

# Ghép chung các vấn đề và quá trình

COMMON PROBLEMS FRAMEWORK ACTIVITIES	<div> <div>communication</div> <div>planning</div> <div>modeling</div> <div>construction</div> </div>											
Software Engineering Tools												
Product Functions												
Text input												
Editing and formatting												
Automatic copy edit												
Page layout capability												
Automatic indexing and TOC												
File management												
Document production												

Figure 21.2 Melding the Problem and the Process

# Dự án

- Dự án gặp rắc rối khi...
  - Người viết phần mềm không hiểu yêu cầu khách hàng của họ.
  - Phạm vi sản phẩm chưa được xác định.
  - Quản lý thay đổi yếu kém
  - Thay đổi công nghệ đã được chọn.
  - Nghiệp vụ bị thay đổi
  - Hạn công việc không thực tế.
  - Người sử dụng cản trở.
  - Mất tài trợ.
  - Thiếu nhân lực có kỹ năng thích hợp.
  - Quản lý(nhân viên) không có thực tiễn và kinh nghiệm tốt nhất.



# Tiếp cận chung tới dự án

- **Mở đầu thuận lợi.** Điều này được thực hiện bằng cách làm việc chăm chỉ (rất cố gắng) để hiểu được vấn đề cần giải quyết. Sau đó thiết lập mục tiêu thực tế và kỳ vọng.
- **Duy trì động lực.** Người quản lý phải khích lệ để giữ biến động nhân sự tối thiểu. Nhóm nên tập trung vào chất lượng trong mọi công việc thực hiện, và người quản lý cấp cao nên làm mọi thứ có thể để không cản trở dự án.
- **Theo dõi tiến trình.** Đối với dự án phần mềm, việc theo dõi quá trình như một phần sản phẩm công việc. Được thực hiện như một phần của việc đảm bảo chất lượng.
- **Đưa ra quyết định thông minh.** Về bản chất, quyết định của người quản lý dự án và nhóm phần mềm nên thật đơn giản.
- **Tiến hành phân tích sau sự án.** Tiến hành cơ chế phù hợp rút ra bài học sau mỗi dự án.

# Để tìm ra bản chất của dự án

- Tại sao hệ thống được phát triển?
- Hệ thống sẽ như thế nào?
- Khi nào hệ thống sẽ được hoàn chỉnh?
- Ai chịu trách nhiệm?
- Công việc sẽ được thực hiện ở đâu?
- Cần bao nhiêu tài nguyên? (ví dụ, con người, các phần mềm, công cụ, cơ sở dữ liệu)

**Barry Boehm [Boe96]**

# Phán đoán thực tế

- Quản lý rủi ro
- Chi phí thực nghiệm và ước lượng kế hoạch
- Quản lý dự án dựa trên Metrics
- Theo dõi giá trị thu được
- Theo dõi thiếu sót với chất lượng đề ra
- Hiểu biết về Quản lý dự án

# Tài liệu tham khảo

- Slide đi kèm với Software Engineering: A Practitioner's Approach, 7/e by Roger S. Pressman
- Slides copyright © 1996, 2001, 2005, 2009 by Roger S. Pressman
- Chỉ dùng cho mục đích giáo dục phi lợi nhuận.
- Có thể sửa đổi slide chỉ nhằm mục đích phục vụ sinh viên đại học trong những môn học liên quan tới sách Software Engineering: A Practitioner's Approach, 7/e. Nghiêm cấm mọi hoạt động sửa đổi khác hoặc sử dụng không được sự cho phép của tác giả.
- Mọi thông tin bản quyền phải được đi kèm nếu những slide này được đăng lên mạng để phục vụ sinh viên.