

# Nhập môn công nghệ phần mềm

## Tiến trình phần mềm

GV: ThS. Ngô Tiến Đức



- 2

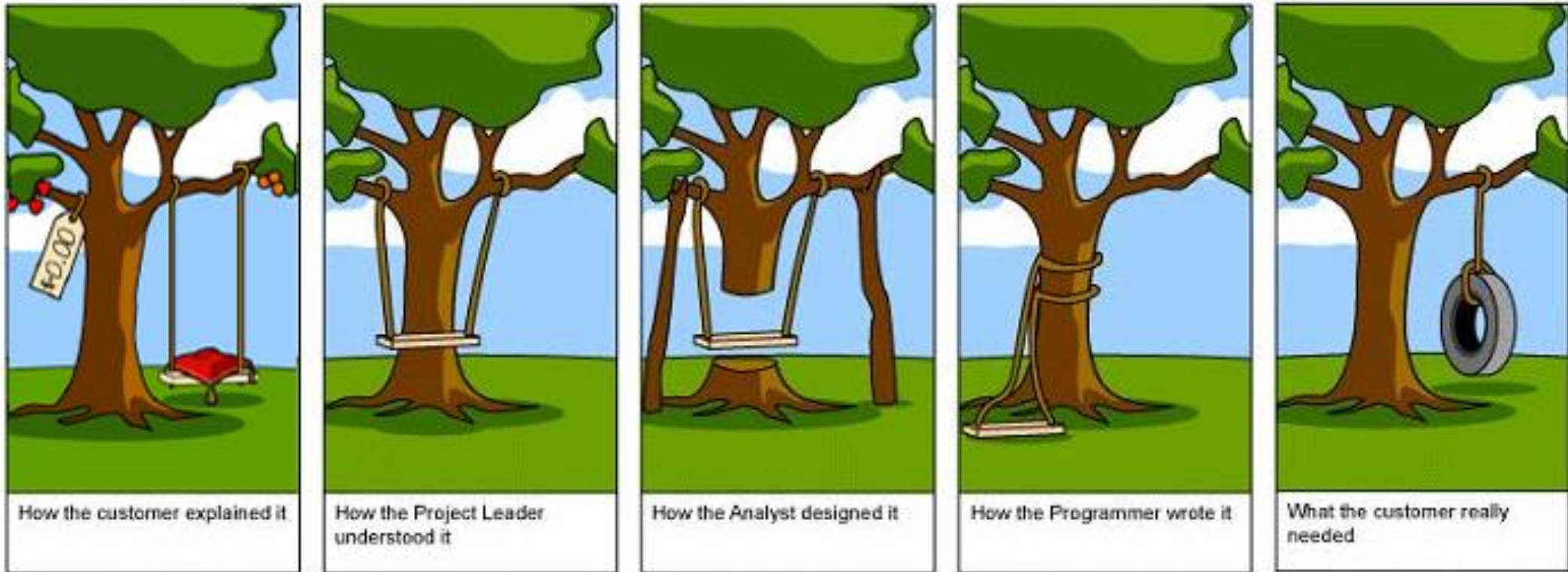
# Nội dung chính (2)

- Tiến trình thống nhất
- Các pha trong tiến trình thống nhất
- Cải tiến tiến trình phần mềm

- 4

# Luồng lấy yêu cầu (1)

Mục tiêu: Xác định nhu cầu của khách hàng



# Luồng lấy yêu cầu (1)

(J4F)





- Cần xác định lĩnh vực phần mềm
- Xây dựng mô hình nghiệp vụ:
  - Làm việc với chuyên gia nghiệp vụ
  - Sử dụng công cụ UML tiếp cận theo hướng nghiệp vụ
  - Đánh giá tính khả thi
    - Khách hàng cảm thấy chi phí không hợp lý?



- Thời hạn bàn giao
- Độ tin cậy
- Chi phí
- Một số yêu cầu khác: độ tin cậy, thời gian phản hồi, portability...



# Luồng phân tích (1)

Mục tiêu:

- Phân tích và làm mịn (refine) yêu cầu của khách hàng
- Mô tả yêu cầu bằng ngôn ngữ kỹ thuật

# Luồng phân tích (2)

- Tài liệu phải thống nhất được cả bên khách hàng và đội phát triển
- Artifact của luồng lấy yêu cầu: Dành cho khách hàng
  - > Tài liệu mô tả bằng ngôn ngữ tự nhiên – không chính xác
- Artifact của luồng lấy phân tích: Dành cho đội phát triển
  - > Tài liệu mô tả bằng ngôn ngữ kỹ thuật – chính xác và khoa học
  - > Cần 2 loại tài liệu đặc tả riêng biệt



- Đầy đủ yêu cầu
- Không có mâu thuẫn
- Không nhập nhằng
- Theo dõi được (Traceability)



- Ước lượng chi phí
- Phân chia công việc
- Ước lượng thời gian
- Xác định các mốc (milestone) và sản phẩm sau từng mốc

# Luồng thiết kế (1)

- Mục tiêu: Mô hình hóa kết quả phân tích cho đến khi lập trình viên có thể code được
- Các yêu cầu phi chức năng:
  - Chọn ngôn ngữ lập trình
  - Tính tái sử dụng
  - Tính khả chuyển

# Luồng thiết kế (2)

Thiết kế cổ điển:

- Thiết kế kiến trúc: Chia nhỏ sản phẩm thành các module
- Thiết kế chi tiết: Lựa chọn cấu trúc dữ liệu và giải thuật

# Luồng thiết kế (3)

Thiết kế hướng đối tượng:

- Các class đã được xác định từ pha phân tích
  - Thiết kế kiến trúc: Xác định quan hệ giữa các class
  - Thiết kế chi tiết: Thiết kế thuộc tính và phương thức cho các class
- > Kết quả cần đạt được: Bản mẫu các class, thuộc tính và phương thức + thuật toán

# Luồng cài đặt

- Mục tiêu: Cài đặt hệ thống theo kết quả từ luồng thiết kế
- Phương pháp: Cài đặt theo class và module -> tích hợp
- Kết quả đạt được: Hệ thống hoàn chỉnh theo yêu cầu





- 17

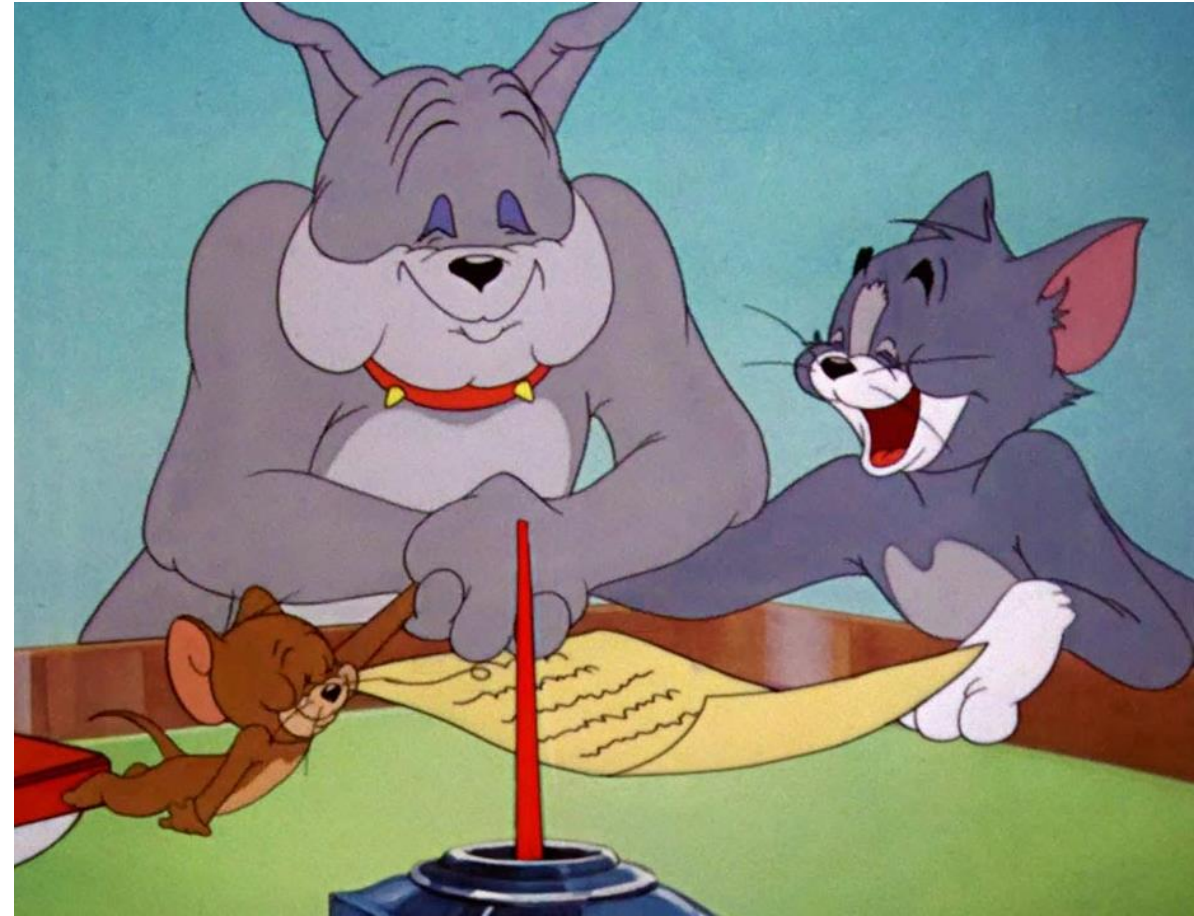


- Quy trình và ký hiệu đồ họa khác nhau
- Ba phương pháp luận hướng đối tượng thành công nhất:

- Booch's method
- Jacobson's OOSE (Object-oriented Software Engineering)
- Rumbaugh's OMT (Object Modeling Technique)

# Tiến trình thống nhất (2)

- 1995: Thống nhất 3 phương pháp luận
- 1997: UML ra đời
- 1999: Công bố Rational Unified Process
  - Unified Software Development Process
  - Unified Process





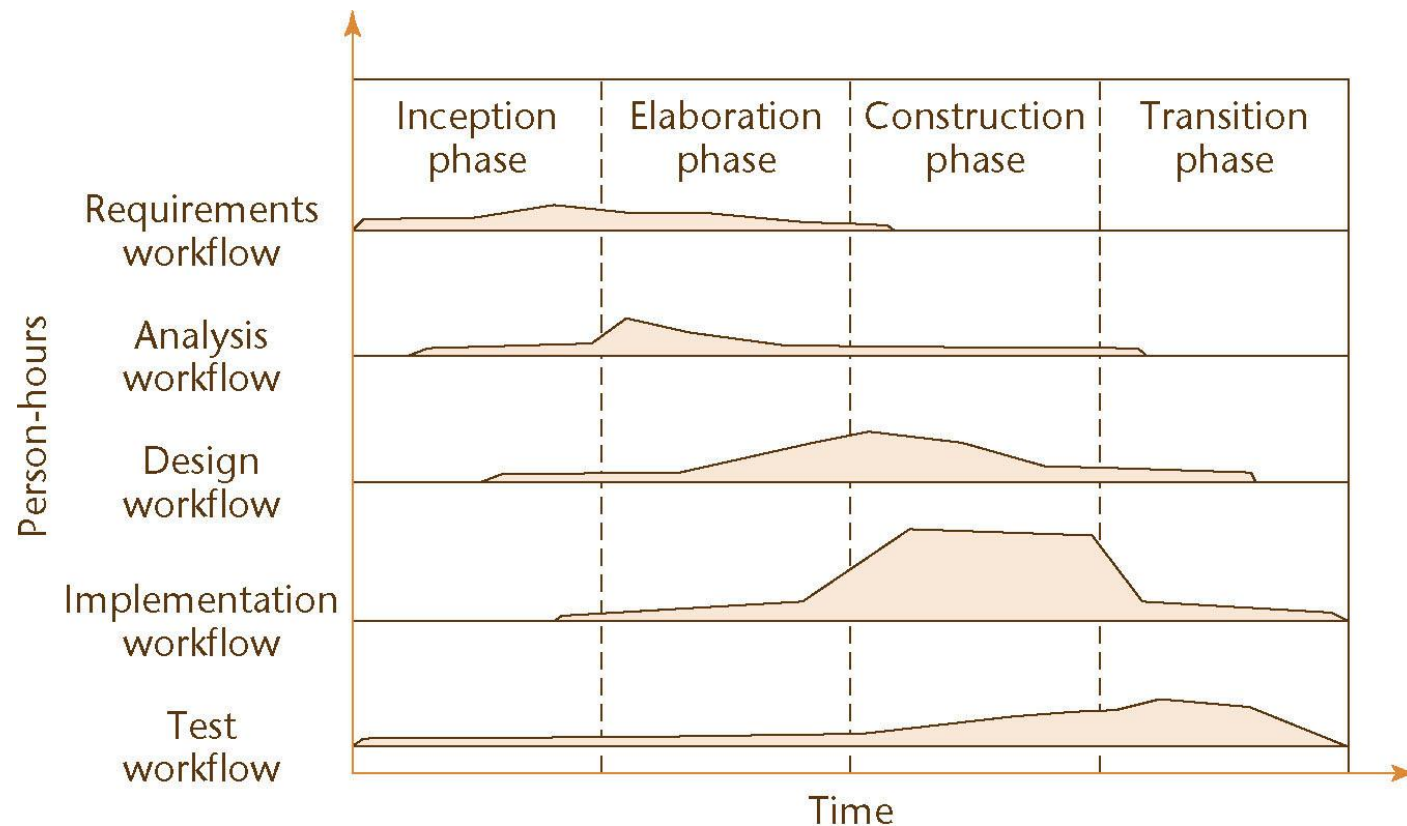
- 20



- Biểu đồ ca sử dụng (Use Case)
- Biểu đồ hoạt động (Activity)
- Biểu đồ tương tác:
  - Biểu đồ tuần tự (Sequence)
  - Biểu đồ giao tiếp/cộng tác (Communication/Collaboration)
- Biểu đồ trạng thái (Statechart)
- Biểu đồ cấu trúc tĩnh (Static Structure)
  - Biểu đồ lớp (Class)
  - Biểu đồ đối tượng (Object)
- Biểu đồ thực thi (Implementation)
  - Biểu đồ thành phần (Component)
  - Biểu đồ triển khai (Deployment)

# Các pha trong tiến trình thống nhất (1)

Quá trình phát triển trải qua 4 bước tăng trưởng, mỗi bước tương ứng với một pha:





- 23



- 24





- 25



- Giá sản phẩm có hợp lý không?
- Bao lâu vốn được quay vòng?
- Nếu từ bỏ dự án thì chi phí thế nào?
- Chiến dịch tiếp thị sản phẩm cho phần mềm COTS?
  - COTS: Commercial-Off-The-Shelf
- Sản phẩm có thể bàn giao đúng hạn không? Nếu muộn thì thiệt hại gì?



- Nhân lực

- Phần cứng hiện tại có đủ tốt không? Có cần phần cứng mới không?

- Công cụ hỗ trợ

- Có đủ các chức năng cần thiết không?

# Pha khởi đầu/đánh giá (4)

Quan hệ với các luồng:

- Bao gồm phần lớn luồng lấy yêu cầu
- Một phần luồng phân tích, thiết kế
- Gần như không bao gồm cài đặt
  - Chỉ khi cần xây dựng bản mẫu khái niệm (proof-of-concept prototype)
- Kiểm thử yêu cầu



- 29

# Pha xây dựng

- Mục tiêu: Xây dựng phiên bản đầu tiên hoạt động được của sản phẩm (bản beta)
- Làm tài liệu hướng dẫn sử dụng cho khách hàng
- Cập nhật bản phân tích kinh doanh (nếu cần)
- Quan hệ với các luồng:
  - Bao gồm thiết kế chi tiết và phần lớn luồng cài đặt
  - Luồng kiểm thử: unit test, integration test, product test



- 31

# Cải tiến tiến trình phần mềm (1)

CMM (Capability Maturity Model)

- Chuẩn quản lý quy trình chất lượng của các sản phẩm phần mềm
- Bộ các tiêu chuẩn đánh giá chiến lược cải tiến tiến trình phần mềm: SW-CMM
- Các tài liệu HDSD đã hoàn chỉnh
- Phiên bản nâng cấp: CMMI (Capability Maturity Model Integration)



# Cải tiến tiến trình phần mềm (2)

## SW - CMM

- Ra đời năm 1986 bởi SEI
- Mục tiêu: Cải tiến tiến trình phần mềm
  - Cải thiện chất lượng
  - Giao hàng đúng hạn, phù hợp ngân sách
  - Cải tiến kỹ thuật
- 5 cấp độ



- Không thể đoán trước toàn bộ tiến trình
- Việc quản lý chỉ bao gồm xử lý các rủi ro gặp phải
- Gần như tất cả các công ty đều đạt chuẩn level 1



- Các quyết định đưa ra dựa vào các dự án trước đó
- Tạo ra các thước đo (metrics)
- Sử dụng để ước lượng cho các dự án tiếp theo
- Sửa lỗi ngay khi phát hiện
- Mất khoảng 3 - 5 năm để từ level 1 -> level 2



- Các khía cạnh quản lý và kỹ thuật được xác định rõ ràng
- Liên tục nâng cao chất lượng và năng suất (productivity)
- Việc đánh giá (reviews) được thực hiện thường xuyên
- Mất khoảng 1.5 đến 3 năm để từ level 2 -> level 3

# Cải tiến tiến trình phần mềm (6)

SW – CMM level 4: Managed

- Chất lượng và năng suất được đặt ra cho mỗi dự án
  - Liên tục được giám sát
- Điều chỉnh dựa theo kết quả thống kê chất lượng



- Tiến trình được cải tiến liên tục:

- Kiểm soát quy trình
- Kiểm soát chất lượng thống kê
- Ghi nhận phản hồi và kinh nghiệm sau mỗi sản phẩm

# Cải tiến tiến trình phần mềm (8)

Key Process Areas cho từng level:

5. Optimizing level: Process control	Defect prevention Technology change management Process change management
4. Managed level: Process measurement	Quantitative process management Software quality management
3. Defined level: Process definition	Organization process focus Organization process definition Training program Integrated software management Software project engineering Intergroup coordination Peer reviews
2. Repeatable level: Basic project management	Requirements management Software project planning Software project tracking and oversight Software subcontract management Software quality assurance Software configuration management
1. Initial level: Ad hoc process	Not applicable



# Bài tập về nhà

Trả lời câu hỏi từ 21 đến 30 trong ngân hàng câu hỏi thi





- 41