



TRƯỜNG ĐH CÔNG NGHỆ ĐÔNG Á
KHOA CNTT - BỘ MÔN CNPM

MÔN HỌC: CÔNG NGHỆ PHẦN MỀM

(IT3226)

Giảng viên: Ths. Nguyễn Thị Nga

Email: ngant2@eaut.edu.vn

Mobile: 0946.046.662



Mục tiêu môn học

- ➡ Hiểu một số mô hình phát triển phần mềm
 - ➡ Phân tích dựa trên sơ đồ dòng dữ liệu (DFD), phân tích hướng đối tượng sử dụng UML
 - ➡ Kiểm tra đánh giá hệ thống
 - ➡ Sử dụng phần mềm hỗ trợ trong mỗi hoạt động trong quản trị dự án.
-



Tài liệu học tập

- ▶ • Giáo trình: R. Pressman, Software Engineering: A practitioner's approach, 8th Edition, McGraw Hill 2016
 - ▶ Bài giảng:
 - ▶ Slides
 - ▶ Lab guides
 - ▶ Tài liệu tham khảo: I. Sommerville, Software Engineering 10th Edition, Addison Wesley 2017
-



NỘI DUNG MÔN HỌC

- 1 Giới thiệu chung về CNPM
 - 2 Phân tích yêu cầu PM và đặc tả HT
 - 3 Kiến trúc phần mềm
 - 4 Thiết kế hệ thống phần mềm
 - 5 Thiết kế giao diện người dùng
 - 6 Kiểm thử phần mềm
 - 7 Quản lý dự án phần mềm
-



Nội Quy và đánh giá

- ▶ Chấp hành nội quy giờ học, nề nếp
 - ▶ Đánh giá:
 - ▶ Điểm chuyên cần: 10% (đi học đầy đủ 7 điểm, làm bài tập, phát biểu 3 điểm)
 - ▶ Bài giữa kỳ: 20%
 - ▶ Thi kết thúc: 70% (hình thức thi: Trắc nghiệm+Tự luận)
-



CHƯƠNG 1: GIỚI THIỆU CHUNG VỀ CÔNG NGHỆ PHẦN MỀM

1.1. Định nghĩa phân loại phần mềm

1.2. Lịch sử phát triển

1.3. Các công cụ CASE

1.4 Các vấn đề trong CNPM



1.1. Định nghĩa phân loại phần mềm

- ▶ Phần mềm là gì?
 - ▶ Phân loại phần mềm
 - ▶ Khái niệm Công nghệ phần mềm
-



Định nghĩa phần mềm

- ▶ Định nghĩa phần mềm (software): Các chương trình, quy trình, và dữ liệu liên quan đến việc vận hành của một hệ thống máy tính.
 - ▶ Định nghĩa 2:
 - ▶ Các lệnh (chương trình máy tính) khi được thực hiện thì cung cấp những chức năng và kết quả mong muốn
 - ▶ Các cấu trúc dữ liệu làm cho chương trình thao tác thông tin thích hợp.
 - ▶ Các tài liệu mô tả thao tác và cách sử dụng chương trình
 - ▶ Phân biệt phần mềm với phần cứng và các khái niệm liên quan.
-



Đặc tính của HW và SW

HW

- ▶ Vật “cứng”
- ▶ Kim loại
- ▶ Vật chất
- ▶ Hữu hình
- ▶ Sản xuất công nghiệp bởi máy móc là chính
- ▶ Định lượng là chính
- ▶ Hỏng hóc, hao mòn

SW

- ▶ Vật “mềm”
- ▶ Kỹ thuật sử dụng
- ▶ Trừu tượng
- ▶ Vô hình
- ▶ Sản xuất bởi con người là chính
- ▶ Định tính là chính
- ▶ Không hao mòn



Phân loại phần mềm

- ▶ Phần mềm hệ thống (System Software): Hệ điều hành, phần mềm điều khiển thiết bị, trình điều khiển, tiện ích hệ thống.
 - ▶ Phần mềm ứng dụng (Application Software): Các chương trình dành cho người dùng cuối (ứng dụng văn phòng, phần mềm kế toán, trình duyệt web...).
 - ▶ Phần mềm nhúng (Embedded Software): Điều khiển các hệ thống nhúng trong thiết bị điện tử. Vật “cứng”
-



Phân loại phần mềm

- ▶ Phần mềm mạng (Network Software): Quản lý mạng và giao tiếp giữa các máy tính.
 - ▶ Phần mềm thương mại và phần mềm nguồn mở: So sánh giữa các mô hình phát triển phần mềm thương mại (Commercial Software) và mã nguồn mở (Open Source Software).
 - ▶ Phần mềm trí tuệ nhân tạo (Artificial Intelligent SW)
-



Đặc điểm của phần mềm

- ▶ Không phải là sản phẩm vật lý.
 - ▶ Tính dễ thay đổi và khả năng phát triển liên tục.
 - ▶ Chu kỳ sống của phần mềm: Đặc điểm, cấu trúc và các giai đoạn chính (phân tích yêu cầu, thiết kế, lập trình, kiểm thử, bảo trì).
-



Định nghĩa công nghệ phần mềm

- ▶ Theo IEEE [1993]: CNPM là
 - ▶ (1) việc áp dụng phương pháp tiếp cận có hệ thống, bài bản và được lượng hóa trong phát triển, vận hành và bảo trì phần mềm;
 - ▶ (2) nghiên cứu các phương pháp tiếp cận được dung trong (1)
 - ▶ Theo Pressman [1995]: CNPM là bộ môn tích hợp cả **quy trình**, các **phương pháp**, các **công cụ** để phát triển phần mềm máy tính
-



Định nghĩa công nghệ phần mềm

- ▶ Sommerville [1995]: CNPM là lĩnh vực liên quan đến **lý thuyết, phương pháp và công cụ** dùng cho phát triển phần mềm
 - ▶ K. Kawamura [1995]: CNPM là lĩnh vực học vấn về các **kỹ thuật, phương pháp luận công nghệ học** (lý luận và kỹ thuật được hiện thực hóa trên những nguyên tắc, nguyên lý nào đó) trong toàn bộ quy trình phát triển phần mềm nhằm nâng cao cả chất và lượng của sản xuất phần mềm
-



Định nghĩa công nghệ phần mềm

- ▶ Định nghĩa: Công nghệ phần mềm là lĩnh vực khoa học về các **phương pháp luận**, **kỹ thuật** và **công cụ** tích hợp trong quy trình sản xuất và vận hành phần mềm nhằm **tạo ra phần mềm** với những chất lượng mong muốn



Các mục tiêu chính của CNPM

- ▶ Tăng **năng suất** và **chất lượng** phần mềm
- ▶ Quản lý **lập lịch** hiệu quả
- ▶ **Giảm chi phí** phát triển phần mềm
- ▶ Đáp ứng **yêu cầu** và **nhu cầu** của khách hàng
- ▶ Tăng cường **quy trình** kỹ nghệ phần mềm
- ▶ Tăng cường **thực hành** kỹ thuật phần mềm
- ▶ **Hỗ trợ** hiệu quả và có hệ thống các hoạt động của kỹ sư phát triển



CNPM là công nghệ phân lớp



- ▶ Quy trình (Process): hệ thống các giai đoạn mà quá trình phát triển phần mềm phải trải qua.
- ▶ Các phương pháp (Methods): phương pháp thực hiện cho từng giai đoạn trong quy trình phát triển phần mềm
- ▶ Các công cụ (Tools): các phương tiện hỗ trợ tự động hay bán tự động cho một giai đoạn nào đó trong quá trình xây dựng phần mềm.



Quy trình- Process

- ▶ Gắn kết các lớp với nhau
 - ▶ Nền tảng cho kỹ thuật phần mềm
 - ▶ Đảm bảo thời gian phát triển
 - ▶ Tạo cơ sở cho việc kiểm soát, quản lý dự án phần mềm
 - ▶ Thiết lập bối cảnh mà các phương pháp kỹ thuật được sử dụng
 - ▶ Tạo sản phẩm
 - ▶ Thiết lập các cột mốc
 - ▶ Đảm bảo chất lượng
 - ▶ Quản lý thay đổi
-



Các phương pháp- Methods

- ▶ Cung cấp **kỹ thuật** cho xây dựng phần mềm
 - ▶ Các tác vụ: giao tiếp, phân tích yêu cầu, mô hình thiết kế, xây dựng chương trình, kiểm thử và hỗ trợ.
 - ▶ Dựa trên các nguyên tắc cơ bản:
 - ▶ Để chi phối từng lĩnh vực công nghệ
 - ▶ Bao gồm các hoạt động mô hình hóa
-



Công cụ- Tools

- ▶ Tự động hoặc bán tự động hỗ trợ cho **quy trình** và các **phương pháp**
 - ▶ Hướng đến chất lượng - A quality focus
 - ▶ Nền tảng
 - ▶ Bất kỳ cách tiếp cận kỹ thuật nào đều phải dựa trên cam kết về chất lượng
 - ▶ Thúc đẩy liên tục việc cải tiến quy trình
-



3 giai đoạn chung của CNPM

- ▶ Pha định nghĩa (Definition phase)
 - ▶ Pha phát triển (Development phase)
 - ▶ Pha hỗ trợ (Support phase)
-



3 giai đoạn chung của CNPM

- ▶ Pha định nghĩa (Definition phase)
 - ▶ Pha phát triển (Development phase)
 - ▶ Pha hỗ trợ (Support phase)
-



Pha định nghĩa

- ▶ Xác định cái gì “**WHAT**”.
 - ▶ Thông tin nào được xử lý,
 - ▶ Chức năng và hiệu quả mong muốn,
 - ▶ Hành vi mong đợi của hệ thống,
 - ▶ Các giao diện cần thiết lập,
 - ▶ Những ràng buộc về thiết kế,
 - ▶ Và những tiêu chí cần thẩm định.
- ▶ Các **yêu cầu chính** của hệ thống và phần mềm được xác định.



Pha phát triển

- ▶ Xác định như thế nào “**HOW**”.
 - ▶ Cách thức dữ liệu được cấu trúc,
 - ▶ Chức năng được triển khai trong kiến trúc phần mềm,
 - ▶ Các chi tiết thủ tục được cài đặt,
 - ▶ Cách xác định các đặc điểm của giao diện,
 - ▶ Cách chuyển từ thiết kế sang lập trình,
 - ▶ Và cách thức kiểm thử.
-



Pha hỗ trợ

- ▶ Liên kết với các thay đổi **“CHANGE”**
 - ▶ Sửa lỗi,
 - ▶ Thích nghi với yêu cầu của môi trường,
 - ▶ Và các thay đổi bởi yêu cầu của khách hàng.
 - ▶ 4 loại thay đổi: Sửa chữa, Thích ứng, Nâng cao, và Phòng ngừa (Correction, Adaptation, Enhancement, and Prevention).
-



1.2. Lịch sử phát triển CNPM

- ▶ Giai đoạn sơ khai (Trước 1960)
 - ▶ Từ khi phần mềm đầu tiên được viết để điều khiển máy tính.
 - ▶ Tầm quan trọng của ngôn ngữ máy và sự phát triển của Assembly.
 - ▶ Sự phát triển của các phương pháp lập trình (1960-1980)
 - ▶ Sự ra đời của các ngôn ngữ lập trình cấp cao như FORTRAN, COBOL.
 - ▶ Sự ra đời của khái niệm phần mềm như một công nghệ độc lập với phần cứng.
 - ▶ Vấn đề về "Khủng hoảng phần mềm" trong những năm 1970.
-



1.2. Lịch sử phát triển CNPM

- ▶ Sự bùng nổ của ngành công nghệ phần mềm (1980 đến nay)
 - ▶ Sự phát triển của các mô hình phát triển phần mềm (Waterfall, Agile, DevOps).
 - ▶ Sự gia tăng của các hệ điều hành, công nghệ đám mây và phần mềm dịch vụ (SaaS).
 - ▶ Vai trò của phần mềm trong thế giới hiện đại (trong thương mại điện tử, truyền thông, trí tuệ nhân tạo).
-



1.3. Các công cụ CASE (Computer-Aided Software Engineering)

- ▶ Công cụ CASE: Các công cụ hỗ trợ phát triển phần mềm tự động hoặc bán tự động.
 - ▶ Phân loại theo mục đích sử dụng:
 - ▶ Upper CASE tools: Công cụ hỗ trợ trong các giai đoạn đầu của phát triển phần mềm (phân tích, thiết kế).
 - ▶ Lower CASE tools: Hỗ trợ trong các giai đoạn sau (lập trình, kiểm thử, bảo trì).
 - ▶ Integrated CASE tools: Công cụ tích hợp cho toàn bộ quy trình phát triển phần mềm.
-



1.3. Các công cụ CASE (Computer-Aided Software Engineering)

- ▶ Vai trò của CASE tools trong phát triển phần mềm:
 - ▶ Tăng hiệu suất, giảm lỗi.
 - ▶ Tạo sự thống nhất giữa các bước trong quy trình phát triển.
 - ▶ Các công cụ CASE phổ biến:
 - ▶ Giới thiệu một số công cụ phổ biến: Rational Rose, Enterprise Architect, Visual Paradigm.
 - ▶ Sử dụng CASE trong thực tiễn (ví dụ, tạo sơ đồ UML, quản lý yêu cầu phần mềm).
-



1.4 Các vấn đề trong CNPM

- ▶ **Không** có phương pháp **mô tả rõ ràng** yêu cầu của khách hàng
 - > Sau khi bàn giao sản phẩm dễ phát sinh những trục trặc (troubles)
- ▶ Với những phần mềm quy mô lớn, **tư liệu đặc tả** cố định
 - > Khó đáp ứng nhu cầu thay đổi của người dùng
- ▶ **Phương pháp luận** thiết kế không nhất quán
 - > Thiết kế theo cách riêng dẫn đến giảm chất lượng phần mềm
- ▶ **Không có chuẩn về việc tạo tư liệu** quy trình sản xuất phần mềm
 - > Đặc tả không rõ ràng sẽ làm giảm chất lượng phần mềm



1.4 Các vấn đề trong CNPM

- ▶ Không kiểm thử tính đúng đắn của phần mềm ở từng giai đoạn mà chỉ kiểm ở giai đoạn cuối và phát hiện ra lỗi
 - > thường bàn giao sản phẩm không đúng hạn
- ▶ Coi trọng việc lập trình hơn khâu thiết kế
 - > giảm chất lượng phần mềm
- ▶ Coi thường việc tái sử dụng phần mềm (software reuse)
 - > giảm năng suất lao động
- ▶ Phần lớn các thao tác trong quy trình phát triển phần mềm do con người thực hiện
 - > giảm năng suất lao động



1.4 Các vấn đề trong CNPM

- ▶ Không chứng minh được tính đúng đắn của phần mềm
 - > giảm độ tin cậy của phần mềm
 - ▶ Chuẩn về một phần mềm tốt không thể đo được một cách định lượng
 - > Không thể đánh giá được một hệ thống đúng đắn hay không
 - ▶ Đầu tư nhân lực lớn vào bảo trì
 - > giảm hiệu suất lao động của nhân viên
-



1.4 Các vấn đề trong CNPM

- ▶ Công việc bảo trì kéo dài
 - > giảm chất lượng của tư liệu và ảnh hưởng xấu đến những việc khác
 - ▶ Quản lý dự án lỏng lẻo
 - > quản lý lịch trình sản xuất phần mềm không rõ ràng
 - ▶ Không có tiêu chuẩn để ước lượng nhân lực và dự toán
 - > làm kéo dài thời hạn và vượt kinh phí của dự án
-