



## Nhập môn **Công nghệ Phần mềm** (Introduction to **Software Engineering**)

**PGS.TS. Phan Huy Khánh**

khanhph29@gmail.com, phkhanh@dut.udu.vn

### Chương 1 Mở đầu

*"One picture is worth more than ten thousand words"*  
Chinese Proverb



## Thời khóa biểu – Hệ Cao đẳng

⌘ Học Kỳ 1, 2012-2013

⌘ Môn học : **Công nghệ phần mềm**

- CBGD : PGS.TS. Phan Huy Khánh
- Lớp ghép **112CNPM201+ 112CNPM1A**
- Ngày bắt đầu 20/08/2012
- Phòng **A202**, tiết **1-4**



2/66



## Mục đích môn học **Công Nghệ Phần Mềm**

- ⌘ Cung cấp kiến thức liên quan đến :
  - Các khái niệm cơ bản về lĩnh vực công nghệ phần mềm (CNPM)
  - Phương pháp, kỹ thuật, công cụ và nguyên tắc trong quá trình xây dựng/phát triển một sản phẩm phần mềm, hay **tiền trình phần mềm** (Software Process), một quy trình công nghiệp
  - Các kỹ năng cơ bản trong quá trình triển khai tiến trình phần mềm
  - Khả năng tham gia xây dựng/phát triển một phần mềm cụ thể trong khuôn khổ một dự án phần mềm (Software Project)
- ⌘ Nội dung môn học trình bày :
  - Các giai đoạn trong một tiến trình phần mềm nói chung
  - Tập trung vào các chủ đề cơ bản :  
**Phân tích yêu cầu, thiết kế hệ thống, thử nghiệm và kỹ thuật đặc tả**

3/66



## Năng lực và kiến thức yêu cầu

- ⌘ Môn học yêu cầu một số kiến thức tiên quyết :
  - Tin học đại cương
  - Cơ sở Lập trình
  - Toán rời rạc – Đảm bảo cơ sở Toán học cho Tin học
- ⌘ Ngoài ra cần đọc thêm, học thêm về :
  - CNTT, internet nói chung
  - Khái niệm dự án, quản lý dự án nói chung, CNTT nói riêng
  - Những vấn đề kinh tế, quản lý xí nghiệp
  - Những vấn đề con người, xã hội, môi trường
  - Các văn bản chính sách, pháp luật của Nhà nước
- ⌘ Một số kỹ năng mềm :
  - Tiếp cận (tự đọc, tự học), trình bày, diễn đạt vấn đề
  - Khả năng điều hành các cuộc họp, thảo luận, làm việc nhóm
  - Khả năng đối phó sự cố, rủi ro...

4/66



## Đánh giá kết quả học tập

- ⌘ Yêu cầu :
  - Hiểu nội dung trình bày trên lớp
  - Thực hiện các bài tập về nhà
  - Khả năng thực hành
- ⌘ Tinh thần thái độ và năng lực học tập
  - Nghe giảng, ghi chép
  - Trả lời câu hỏi và đặt câu hỏi
  - Tham khảo tài liệu, truy cập internet
  - Tham gia học nhóm, tập thảo luận và thuyết trình
  - Khả năng tìm hiểu những vấn đề thực tiễn và ứng dụng CNTT
- ⌘ Kiểm tra giữa kỳ, cuối kỳ :
  - Thi viết (60 phút)


5/66



## Tài liệu tham khảo


- ⌘ Bài giảng PPT :
  - [1] *Nhập môn Công nghệ Phần mềm*, PGS.TS. Phan Huy Khánh, Khoa CNTT, Tr. ĐHBK, ĐHDN
- ⌘ Sách tiếng Anh :
  - [1] Roger S. Pressman. *Software Engineering A practitioner's Approach*. McGraw-Hill, Six Edition, 2005 (Ebook)
  - [2] Ian Sommerville. *Software Engineering*. McGraw-Hill, 7th Edition, 2007 (Bản dịch của Ngô Trung Việt)
- ⌘ Sách tiếng Việt :
  - [3] Lê Đức Trung. *Công Nghệ Phần Mềm*. Nhà XB Khoa Học Kỹ thuật, Hà Nội 2002
  - [4] Lương Mạnh Bá. *Cơ Sở Công Nghệ Phần Mềm*. Nhà XB Khoa Học Kỹ thuật, Hà Nội 2010
- ⌘ Internet : <http://tailieu.vn/>  
**Lưu trữ** với từ khóa tìm kiếm **lq công nghệ phần mềm**


6/66



## Nội dung môn học


- Chương 1 **Mở đầu**
- Chương 2 Phân tích hệ thống
- Chương 3 **Thiết kế hệ thống**
- Chương 4 **Kiểm thử phần mềm**
- Chương 5 **Đặc tả phần mềm**



7/66



## Chương 1 Mở đầu




- ✂ Một số khái niệm và thuật ngữ
- ✂ Các phương diện của công nghệ phần mềm
- ✂ Những nội dung cơ bản của CNPM
- ✂ Tiến trình phần mềm
- ✂ Kỹ thuật thể hệ 4


8/66



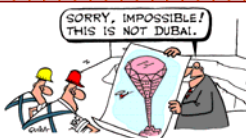

## Computer = Hardware + Software


**Software**






**Hardware/Software Interface**

**Hardware**

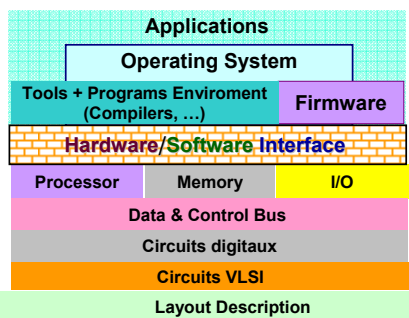




9/66




## Computer Architecture


- ✂ Tổ chức hệ thống theo lớp, nhiều mức trừu tượng



**Software**


**Hardware**


10/66





## Phần mềm (Software)

- ✂ Phần mềm, còn được gọi là (đgl) **nhu liệu**, là :
  - Tập hợp các chương trình, bao gồm, hay tập hợp các câu lệnh
  - Được viết bằng một, hoặc nhiều ngôn ngữ lập trình
  - Thực hiện (chạy) theo một trật tự xác định (thuật toán), sử dụng các đối tượng dữ liệu có tổ chức
  - Tự động thực hiện một số chức năng hoặc giải quyết một bài toán/vấn đề nào đó
- ✂ Đặc điểm của phần mềm là tính trừu tượng và vô hình




Từ những năm 1980, phần mềm được bày bán như là sản phẩm


11/66



## Ví dụ phần mềm

- ✂ Chức năng của một phần mềm :
  - Làm công cụ trợ giúp **người sử dụng** (NSD) thực hiện công việc của mình trong một lĩnh vực chuyên môn/ngành nghiệp trên máy tính
  - Có chất lượng hiệu quả, tốt hơn (một cách tự động) so với công việc thủ công
- ✂ Phần mềm **Quản lý giáo vụ** trong Giáo dục & Đào tạo :
  - NSD làm công tác giáo vụ :
    - ✦ Xếp lớp, xếp ngành, lập thời khoá biểu
    - ✦ Theo dõi, quản lý kết quả học tập
  - NSD là giảng viên :
    - ✦ Đăng ký giảng dạy
    - ✦ Xem thời khoá biểu
  - NSD là Sinh viên :
    - ✦ Đăng ký học phần
    - ✦ Xem điểm, kết quả học tập


12/66



## Công nghệ phần mềm (CNPM) là gì ?

☞ Từ điển *Computer Dictionary* của Microsoft Press®, 1994 :

- **Software Engineering** : The Design and Development of Software (Computer Program), from Concept through Execution and Documentation

☞ Từ điển Larousse 1996 định nghĩa chi tiết hơn :

- Là tập hợp các phương pháp, mô hình, kỹ thuật, công cụ và thủ tục xây dựng một Sản Phẩm Phần Mềm (SPPM)
- Các giai đoạn xây dựng SPPM : *đặc tả* (Specification), *thiết kế* (Design), *lập trình* (programming), *thử nghiệm* (Testing), *sửa sai* (Debugging), *cài đặt* (Setup) để đem vào *ứng dụng* (Application), *bảo trì* (Maintenance) và *lập hồ sơ* (Documentation) đóng gói SP

13/66



## Bách khoa toàn thư mở Wikipedia

☞ Công nghệ (kỹ nghệ) phần mềm :

Vận dụng thực tế những kiến thức khoa học trong thiết kế, xây dựng phần mềm cũng như tài liệu liên quan trong quá trình phát triển, hoạt động và bảo dưỡng

☞ Theo Edsger Dijkstra :

*Khi máy tính điện tử chưa xuất hiện, việc lập trình chưa có khó khăn gì cả*

*Khi đã xuất hiện một vài chiếc máy tính với chức năng kém thì lập trình đã gặp một vài khó khăn nhỏ*

*Giờ đây khi chúng ta đã có những chiếc máy tính khổng lồ thì những khó khăn ấy đã trở nên rất lớn*

*Như vậy, ngành công nghiệp điện tử đã không giải quyết khó khăn nào mà họ chỉ tạo ra những khó khăn mới*

*Họ đã tạo ra khó khăn chính là sử dụng sản phẩm của họ*

14/66



## Vai trò của Công nghệ Phần mềm

☞ CNPM :

- Một ngành không thể thiếu được trong lĩnh vực CNTT&TT (ICT)
- Ngày càng có nhiều hệ thống khác nhau, trong mọi lĩnh vực, được kiểm soát bởi phần mềm

☞ Xu thế ngày nay :

- Con người có thể sản xuất các hệ thống PM lớn và hữu ích là nhờ phần lớn vào sự phát triển của CNPM
- Mọi quốc gia phát triển đều phụ thuộc chủ yếu vào các hệ thống PM có chất lượng
- Xây dựng, bảo trì một hệ thống PM hiệu quả là yêu cầu cần thiết đối với nền kinh tế toàn cầu và của từng quốc gia

15/66



## Đặc điểm của CNPM

☞ Sự khác biệt giữa CNPM và Khoa học Máy tính (KHMT)

- KHMT nghiên cứu lý thuyết và những vấn đề cơ bản về CNTT
- Khi CNPM phát triển mạnh mẽ, các lý thuyết của KHMT vẫn không đủ để đóng vai trò là nền tảng hoàn thiện cho CNPM

☞ Sự khác biệt giữa CNPM và Công nghệ Hệ thống (CNHT)

- CNHT (hay kỹ nghệ hệ thống) liên quan tới mọi khía cạnh của quá trình phát triển hệ thống dựa máy tính bao gồm : phần cứng, phần mềm, và công nghệ xử lý
- Kỹ sư hệ thống phải thực hiện việc đặc tả hệ thống, thiết kế kiến trúc hệ thống, tích hợp và triển khai hệ thống

16/66



## Mục đích của CNPM

☞ Mục đích của CNPM là sản xuất phần mềm có chất lượng

☞ Chất lượng phần mềm :

- Không là một khái niệm đơn giản
- Bao gồm nhiều yếu tố phức hợp

☞ Người ta thường đánh giá theo hai kiểu chất lượng :

- Những yếu tố **chất lượng bên ngoài** (mục đích cuối cùng, chẳng hạn chạy nhanh, dễ hiểu, dễ sử dụng...)
- Những yếu tố **chất lượng bên trong** (chỉ những người làm Tin học chuyên nghiệp mới biết được, chẳng hạn có tính cấu trúc, dễ đọc, dễ bảo trì phát triển...)

☞ Yếu tố chất lượng bên trong là mẫu chốt để đạt được những yếu tố chất lượng bên ngoài

17/66



## Chất lượng bên ngoài của SPPM

☞ Những yếu tố chất lượng bên ngoài của SPPM chủ yếu phục vụ cho người sử dụng (NSD)

☞ Chất lượng bên ngoài gồm :

- Tốc độ nhanh
- Chạy ổn định
- Có tính dễ hiểu, dễ sử dụng (tiếng Anh, tiếng Việt)
- Có tính dễ thích nghi với những thay đổi (tính mở rộng)
- Có tính công thái học (Ergonomics, Human Factor)
- v.v...

18/66



## Chất lượng bên trong của SPPM

- ⌘ Những yếu tố chất lượng bên trong của một SPPM (cho những người làm Tin học chuyên nghiệp) là :
  - Tính đúng đắn
    - ✦ Khả năng thực hiện chính xác công việc đặt ra
  - Tính bền vững
    - ✦ Có thể hoạt động trong những điều kiện bất thường
  - Tính có thể mở rộng
    - ✦ Khả năng dễ sửa đổi để thích nghi với những thay đổi mới
  - Tính sử dụng lại
    - ✦ Khả năng sử dụng lại toàn bộ hay một phần của hệ thống cho những ứng dụng mới
  - Tính tương thích
    - ✦ Có thể dễ dàng kết hợp với các sản phẩm phần mềm khác

19/66



## Các yếu tố chất lượng khác

- ⌘ Hiệu quả đối với nguồn tài nguyên của MTĐT như :
  - Bộ xử lý
  - Bộ nhớ các loại
  - Các thiết bị CNTT&TT...
- ⌘ Tính thích nghi :
  - Dễ sử dụng
  - Dễ chuyển đổi (không phụ thuộc vào cấu hình phần cứng)
  - Dễ kiểm chứng
  - An toàn (được bảo vệ quyền truy nhập)
  - v.v...

20/66



## Phân loại phần mềm

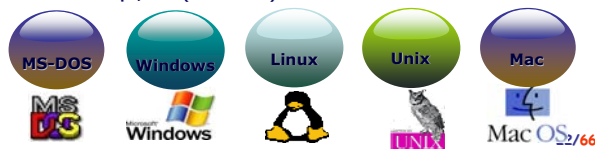
- ⌘ Phần mềm (PM) máy tính là thuật ngữ chung chỉ các chương trình chạy trên máy tính
- ⌘ Có nhiều cách phân loại PM, nhưng thường được phân loại theo phương thức hoạt động, gồm 4 loại :
  - Phần mềm hệ thống (System SoftWare)
  - Phần mềm ứng dụng (Application SoftWare)
  - Phần mềm tiện ích (Utility SoftWare)
  - Phần mềm nhúng (Embedded Software)

21/66



## Phần mềm hệ thống

- ⌘ Chức năng dùng để vận hành máy tính :
  - Giúp điều khiển phần cứng máy tính, các thiết bị chuyên dụng
  - Là nền tảng cho các ứng dụng khác nhau hoạt động
- ⌘ Ví dụ :
  - Các hệ điều hành máy tính Windows XP, Linux, Unix...
  - Các thư viện liên kết động DLL (Dynamic Linked Library)
  - Các trình điều khiển (Driver)
  - Phần sụn/dẻo (FirmWare) và BIOS



## Phần mềm ứng dụng

- ⌘ Chức năng dùng để thực hiện một hay nhiều tác vụ nào đó trên máy tính một cách hiệu quả, nhanh chóng và dễ dàng
- ⌘ Ví dụ :
  - Phần mềm văn phòng (Microsoft Offices, Lotus 1-2-3, FoxPro)
  - Phần mềm đa phương tiện (Multimedia Software)
  - Phần mềm doanh nghiệp BI, ERP, CRM...
  - Phần mềm giáo dục, trợ giúp giảng dạy (e-Learning)
  - Phần mềm Cơ sở dữ liệu (CSDL) và quản trị CSDL
  - Phần mềm đồ họa (Graphics Software), thiết kế (CAD, Designers...)
  - Các hệ chuyên gia, trí tuệ nhân tạo, người máy, v.v...
  - Phần mềm trò chơi (Games)...

23/66



## Phần mềm tiện ích

- ⌘ Chức năng dùng để quản lý và tối ưu hóa mọi máy tính một cách hiệu quả, nhanh chóng và dễ dàng
- ⌘ Ví dụ :
  - Các ngôn ngữ lập trình (Programming Languages)
  - Các phần mềm chuyển dịch mã (Translator) : biên dịch (Compiler), thông dịch (Interpreter)
  - Quản lý đĩa cứng (Disk Defragmenters)
  - Quản lý hiện trạng (System Profilers)
  - Phòng chống virus, hacker (Virus Scanners)
  - Nén giải nén DL (Compression)
  - Tối ưu hệ thống (Registry Cleaners)...

24/66



## Phần mềm nhúng

Hay **hệ thống nhúng** (Embedded System) :

⌘ Phục vụ giải quyết các bài toán chuyên dụng trong nhiều lĩnh vực khác nhau :

- Công nghiệp
- Tự động hoá
- Điều khiển
- Quan trắc
- Truyền tin...

⌘ Mục đích :

- Giúp các sản phẩm điện tử có chức năng hoàn hảo, phục vụ hiệu quả các nhu cầu của NSD với độ an toàn, bảo mật cao

25/66



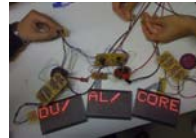
## Ví dụ phần mềm nhúng

⌘ Là những thiết bị cầm tay nhỏ gọn :

- Đồng hồ kĩ thuật số
- Máy chơi nhạc MP3...

⌘ Hoặc là những sản phẩm lớn như :

- Đèn giao thông
- Bộ kiểm soát hoạt động trong nhà máy
- Các hệ thống kiểm soát máy năng lượng hạt nhân
- Dự báo, cảnh báo thời tiết, mưa bão, động đất, sóng thần...



Phần mềm điều khiển mạch LED



Robot của NASA

26/66



## Đặc điểm phần mềm nhúng

⌘ Đặc điểm của các hệ thống nhúng là :

- Được tích hợp vào trong các IC của thiết bị điện tử
- Có khả năng tự trị, chạy trong các thiết bị mà không cần hệ điều hành
- Hoạt động ổn định
- Có tính năng tự động hoá cao

⌘ Ưu điểm :

- Nhỏ gọn
- Dễ cài đặt, đóng gói
- Có thể sản xuất hàng loạt



Network Embedded Module

27/66



## Thuộc tính của SPPM

⌘ Thuộc tính của một SPPM biến đổi tùy theo phần mềm :

- Xuất hiện từ khi được cài đặt và được đưa ra dùng
- Không bao gồm các dịch vụ được cung cấp kèm theo

⌘ Một số thuộc tính quan trọng gồm:

- Khả năng bảo trì để thỏa mãn yêu cầu của khách hàng
- Khả năng tin cậy, an toàn và bảo mật của phần mềm không tạo ra thiệt hại vật chất hay kinh tế trong trường hợp hư hỏng
- Độ hữu hiệu không thể phi phạm các nguồn tài nguyên như bộ nhớ và các chu kì vi xử lý
- Khả năng sử dụng, có một giao diện tương đối dễ cho người dùng và có đầy đủ hồ sơ về phần mềm

28/66



## Tính chất của CNPM/SPPM

⌘ CNPM :

- Tạo ra SPPM "vật chất" thấy được, sao chép được nhưng không "sờ thấy" được
- Hạ tầng cơ sở chủ yếu là điện và viễn thông
- Người ta có thể làm việc từ xa, làm việc "tại gia", mọi lúc mọi nơi...
- Lĩnh vực luôn đòi hỏi tính năng động, tính sáng tạo và khôn ngoan

⌘ SPPM :

- Không có "hàng giả", chỉ bị sao chép "lậu", "ăn cắp"...
- Không bị "mòn cũ", nhưng luôn được "mới hơn" qua phiên bản
- Được phân phối và tiêu thụ theo nhiều phương thức, có thể "vượt biên" không qua khâu kiểm tra Hải quan
- Không phụ thuộc và không bị ảnh hưởng (hoặc chỉ bị rất ít) các yếu tố địa lý, khí hậu, môi trường
- Mở ra các thị trường rất lớn, đầy tiềm năng, cuộc cạnh tranh quyết liệt

29/66



## Tính cạnh tranh của PM

⌘ Cạnh tranh của PM (Software Competitiveness)

- Mỹ, Tây Âu và Đông Âu ngày nay có công nghiệp PM ổn định
- Châu Á có các quốc gia có nền CNPM phát triển : Nhật Bản, Hàn Quốc, Singapore, Ấn Độ, Trung Quốc

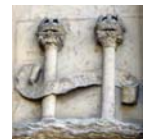
⌘ Bản chất trí tuệ của cạnh tranh :

Innovative, Vital and Successful

Software Product = Human Thought

Need  and HQP (High Qualified Personal)

Human Resource Need for the VN IT Industry

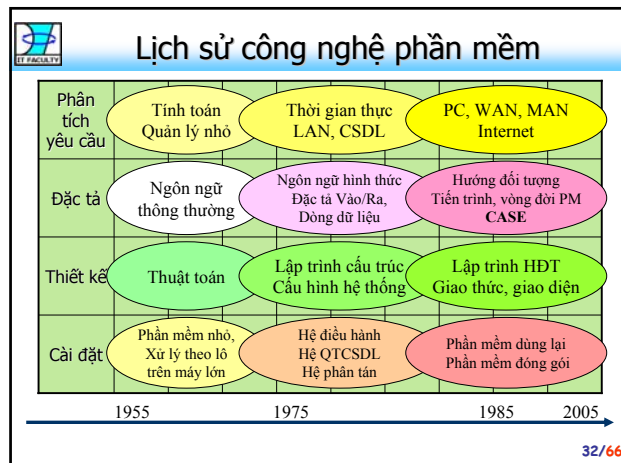


30/66



Tóm tắt quá trình tiến hóa của SPPM	
Thời kỳ đầu tiên 1950 - 1960	Xử lý theo lô (Batch processing) Phần mềm được viết theo đơn đặt hàng
Thời kỳ thứ hai 1960 - 1970	Đa người dùng (Multiusers) Thời gian thực (Real time) Cơ sở dữ liệu (Database) Phần mềm sản phẩm
Thời kỳ thứ ba 1970 - 1990	Hệ thống xử lý phân bố (Distributed processing system) Thông minh (Intelligence) Phần cứng giá thành hạ Hiệu quả tiêu thụ
Thời kỳ thứ tư 1990 trở đi	Hệ thống đồ bàn (Desktop - Personal - Notebook computers) Lập trình hướng đối tượng (Object oriented programming) Lập trình trực quan (Visual programming) Hệ chuyên gia (Expert system) Mạng thông tin toàn cầu (Worldwide communication network) Xử lý song song (Parallel processing) ...

31/66



32/66

### Phát triển phần mềm

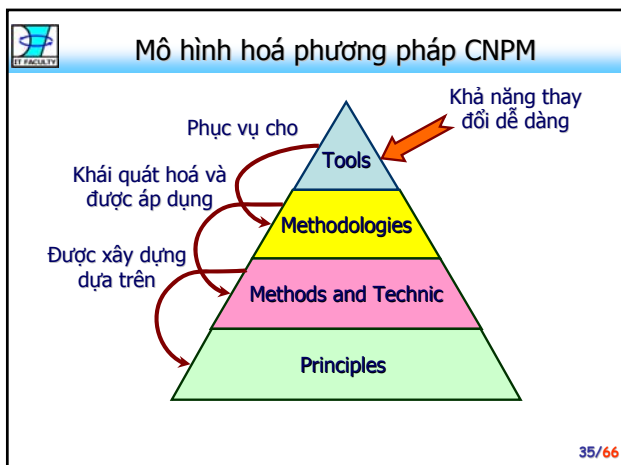
- ❖ Phát triển phần mềm trong khuôn khổ một dự án Tin học :
  - Nhiều người lập trình tham gia, được chia thành nhóm
  - Mỗi nhóm phụ trách giải quyết một phần của dự án
  - Người phụ trách dự án phân bổ công việc cho từng nhóm
  - Các nhóm phải đảm bảo mỗi liên lạc và chia sẻ thông tin
  - Kiểm tra tiến trình phát triển của dự án
  - Kiểm tra chất lượng của SPPM khi hoàn tất
- ❖ Bản chất phát triển phần mềm :
  - Không chỉ là việc lập trình thuần tuý
  - Là triển khai các giai đoạn theo các quan điểm khác nhau
  - Hiện nay thường sử dụng quan điểm theo lớp

33/66

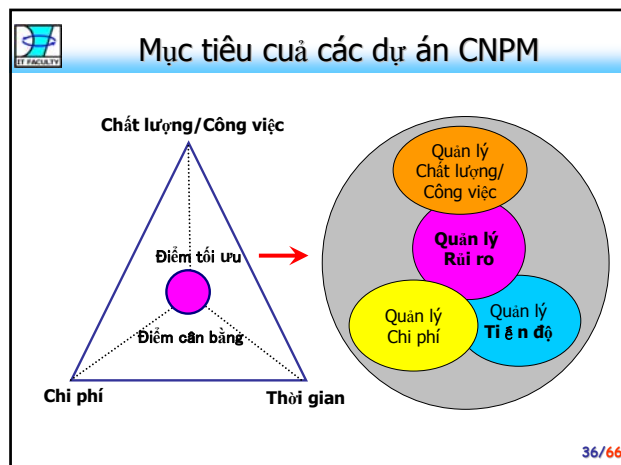
### Nguyên lý, phương pháp, kỹ thuật và công cụ

- ❖ Để xây dựng một HTPM, người ta cần nguyên lý, phương pháp, kỹ thuật và công cụ (tương tự nấu một món ăn)
  - Nguyên lý : tuân theo các quy tắc, mục đích  
Ví dụ : các quy tắc về vệ sinh, an toàn thực phẩm
  - Kỹ thuật : cách thức hay thủ thuật thực hiện công việc  
Ví dụ : Kỹ thuật nấu nướng, chế biến hay bày món ăn
  - Phương pháp hay công nghệ : tập hợp các kỹ thuật  
Ví dụ : Phương pháp nấu món "cá lóc hấp muối"
  - Công cụ : các phương tiện và nguồn tư vấn  
Ví dụ : xoong, nồi, bát đĩa hay thực phẩm
- ❖ Khái niệm tiến trình được hiểu là các giai đoạn triển khai
- ❖ Tính thực dụng của HTPM : khả năng ứng dụng thực tiễn

34/66



35/66



36/66



## Các giai đoạn tiến trình phần mềm

⌘ Gồm 3 giai đoạn chính :

- Giai đoạn xác định
- Giai đoạn phát triển
- Giai đoạn bảo trì

⌘ Các giai đoạn trong tiến trình phần mềm thường sử dụng các công cụ đặc tả (Specification) :

- Đặc tả phi hình thức (Informal Specification)
- Đặc tả hình thức (Formal Specification)



37/66



## Giai đoạn xác định

⌘ Trả lời câu hỏi liên quan đến :

- Dữ liệu (thông tin) cần xử lý
- Mục đích chức năng và môi trường phát triển
- Hai loại câu hỏi :
  - ✦ Là cái gì ? Là gì ? (What?)
  - ✦ Khi nào ? (When?)

⌘ Giai đoạn xác định gồm 3 bước :

- Phân tích hệ thống/ Phân tích hiện trạng
- Lập kế hoạch dự án phần mềm
- Phân tích yêu cầu thực tiễn

38/66



## Giai đoạn phát triển

⌘ Giai đoạn phát triển trả lời câu hỏi :

- Làm như thế nào ? (How?)

⌘ Gồm 3 bước :

- Thiết kế phần mềm :
  - ✦ Sử dụng các công cụ đặc tả và lập trình cấu trúc
- Chọn công cụ hoặc các ngôn ngữ lập trình để tiến hành viết chương trình
- Kiểm thử (phát hiện sai sót, nhầm lẫn...)

39/66



## Giai đoạn bảo trì

⌘ Giai đoạn bảo trì tập trung vào các thay đổi (Modify)

⌘ Có 3 kiểu thay đổi :

- Sửa đổi :
  - ✦ Dù phần mềm có chất lượng tốt, vẫn tồn tại những khiếm khuyết từ việc sử dụng của khách hàng (NSD)
  - ✦ Bảo trì sửa đổi làm thay đổi phần mềm, khắc phục khiếm khuyết
- Thích nghi :
  - ✦ Nhằm làm phần mềm thích nghi với môi trường phần cứng Ví dụ CPU, OS, các thiết bị ngoại vi...
- Nâng cao :
  - ✦ Khách hàng tìm ra những chức năng phụ của phần mềm
  - ✦ Bảo trì hoàn thiện để mở rộng phần mềm ra ngoài những chức năng vốn có

40/66



## Chu kỳ sống của phần mềm

⌘ Chu kỳ sống, hay vòng đời (Life Cycle), của phần mềm là cách thức triển khai, vận dụng thực tiễn một tiến trình phần mềm để có được SPPM

⌘ Có nhiều mô hình khác nhau để thể hiện chu kỳ sống, thường có 4 loại :

- Mô hình thác nước kiểu cổ điển
- Mô hình chữ V, hay chữ W
- Mô hình xoắn tròn ốc
- Mô hình kiểu bản mẫu

41/66

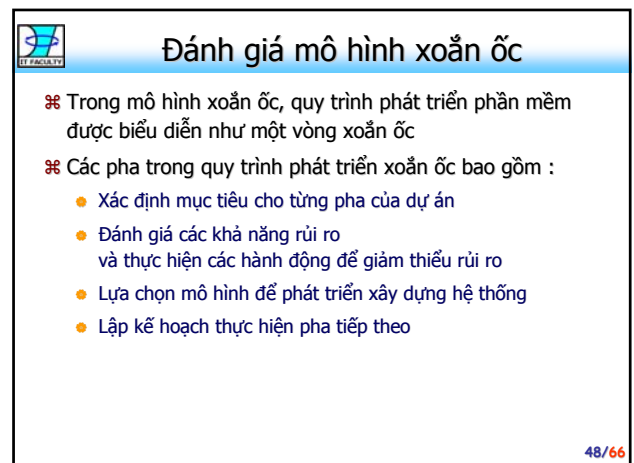
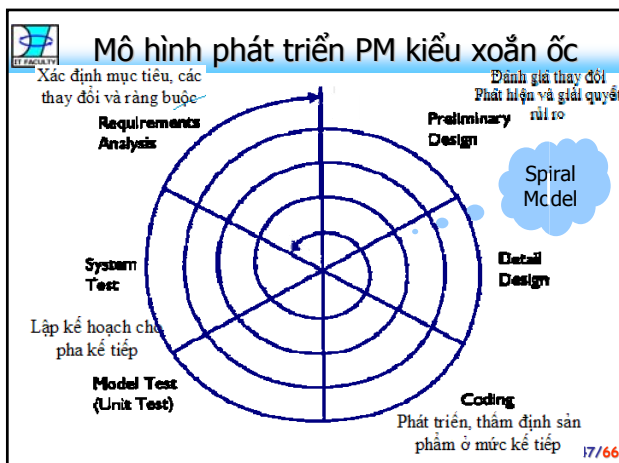
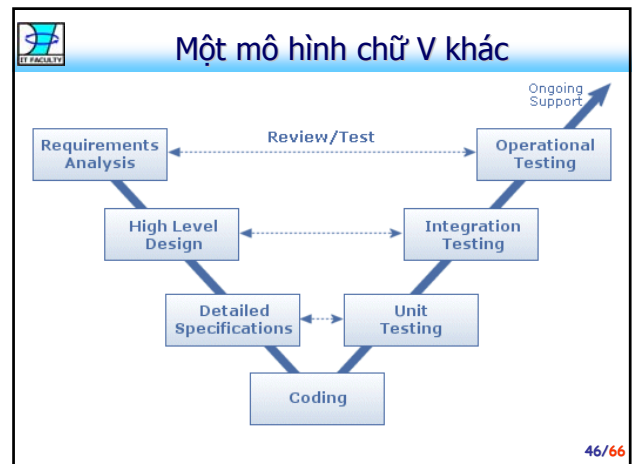
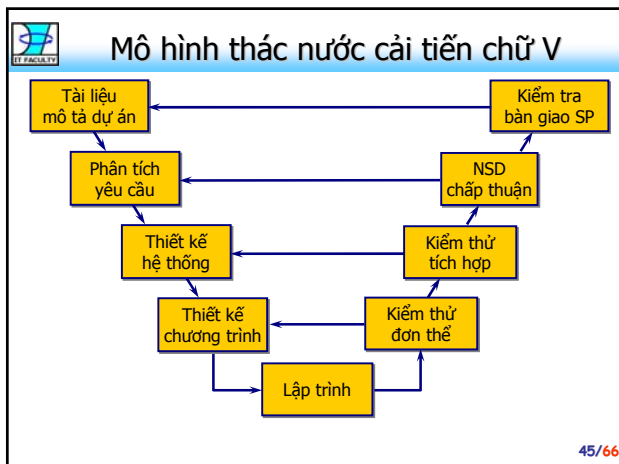
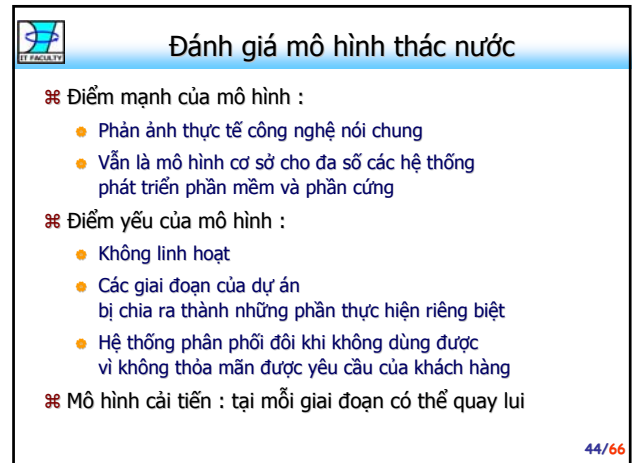
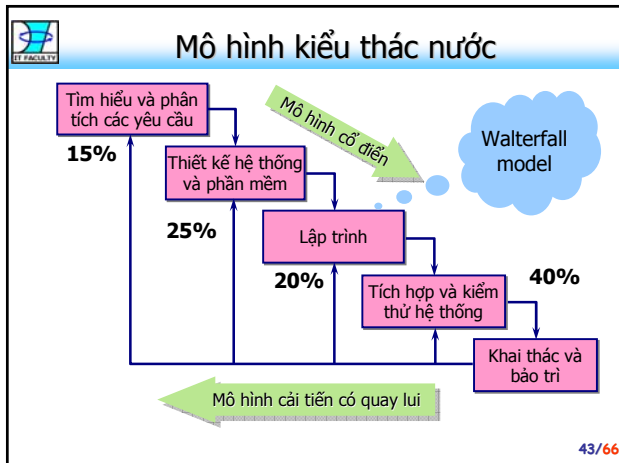


## Chu kỳ sống kiểu cổ điển

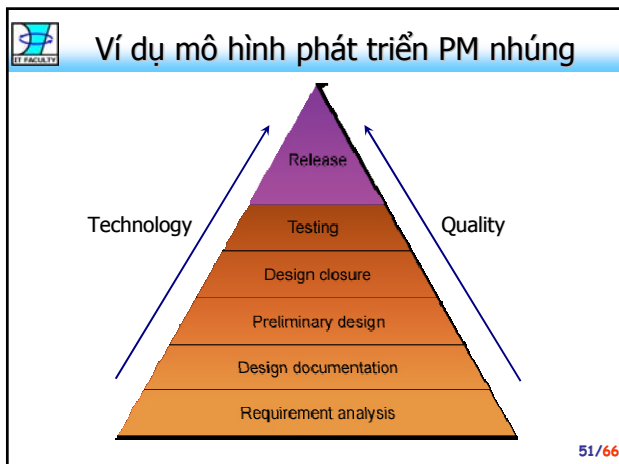
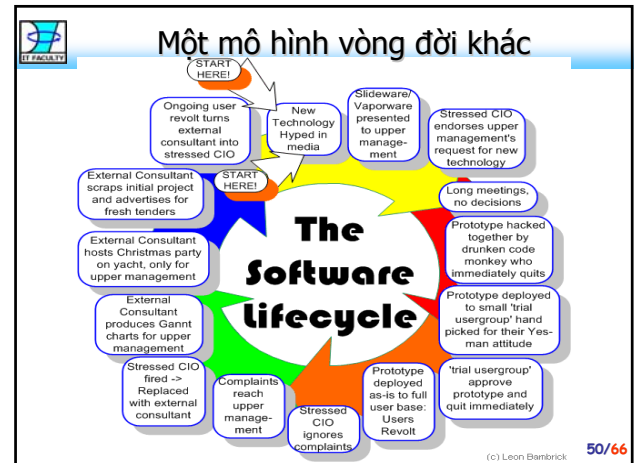
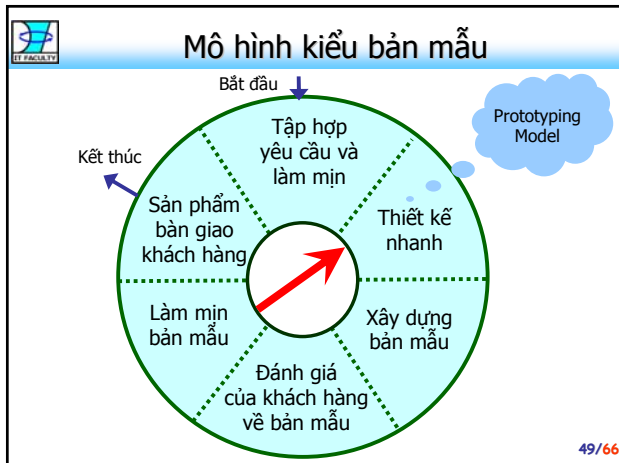
⌘ Chu kỳ sống kiểu cổ điển theo mô hình thác nước ("Waterfall" Model) gồm các giai đoạn như sau :

- Tìm hiểu và phân tích các yêu cầu (RAD - Requirements analysis and definition)
- Thiết kế hệ thống và phần mềm (SSD - System and software design)
- Cài đặt và kiểm thử từng phần (IUT - Implementation and Unit testing)
- Tích hợp và kiểm thử hệ thống (IST - Integration and system testing)
- Khai thác và bảo trì (DEM - Development and Maintenance)

42/66



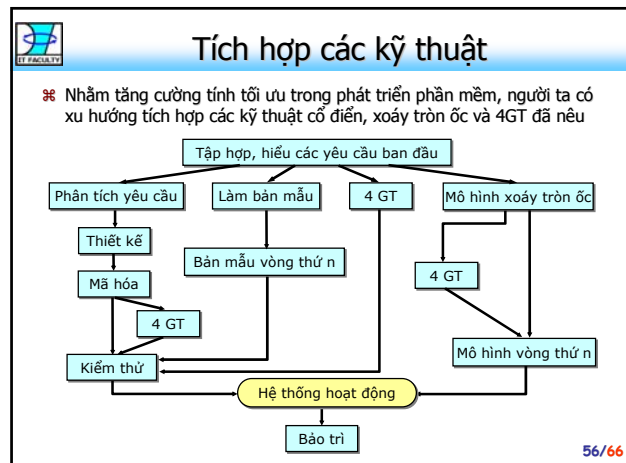
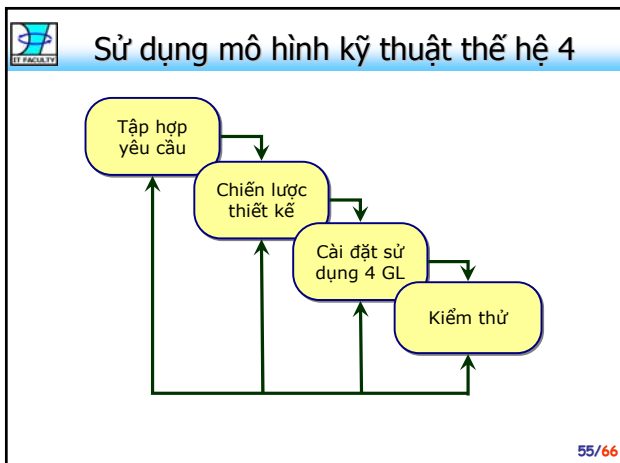




- Các phương pháp CNPM**
- ✂ Phương pháp CNPM bao gồm :
    - Các mô hình hệ thống
    - Các kỹ pháp, quy tắc, hướng dẫn thiết kế
    - Quy trình để xây dựng phần mềm một cách dễ dàng, đảm bảo chất lượng cao và chi phí hiệu quả
  - ✂ Một số phương pháp CNPM đã được đề xuất :
    - Phân tích hướng cấu trúc : Tập trung xác định các chức năng cơ bản của hệ thống
    - Phương pháp hướng đối tượng : Tập trung định nghĩa các đối tượng, các lớp đối tượng và sự cộng tác giữa chúng
  - ✂ Kỹ thuật thể hệ 4 : phương pháp CASE
- 52/66

- Kỹ thuật thể hệ 4 : CASE**
- ✂ Kỹ thuật thể hệ 4 (4th Generation Technology)
    - Sử dụng các công cụ phần mềm CASE (Computer Aided for Software Engineering) phục vụ tiến trình phần mềm
    - CASE tự động sản sinh mã chương trình gốc theo nhu cầu của người phát triển
  - ✂ Có hai loại CASE :
    - Upper-CASE : trợ giúp đặc tả yêu cầu, thiết kế
    - Lower-CASE : trợ giúp lập trình, gỡ lỗi và kiểm thử
- 53/66

- Các thành phần của CASE**
- ✂ CASE thường có các thành phần :
    - Ngôn ngữ phi thủ tục (Non Procedural Language) để truy cập cơ sở dữ liệu, chẳng hạn SQL
    - Bộ sản sinh báo cáo (Report Generator)
    - Bộ thao tác dữ liệu
    - Bộ tương tác và thiết kế màn hình
    - Bộ sinh chương trình (Code Generator)
    - Bảng tính
    - Các công cụ đồ họa...
- 54/66



### Các chi phí trong CNPM

⌘ Chi phí **xây dựng SPPM**, **sử dụng PM** chiếm một phần GDP đáng kể ở tất cả các nước phát triển :

- Chiếm chi phí lớn hơn của cả hệ thống máy tính nói chung
- Trên máy PC thường lớn hơn chi phí phần cứng
- Chi phí bảo trì lớn hơn chi phí xây dựng PM :
- Đối với chi phí xây dựng PM :
  - ✦ Xấp xỉ 60% chi phí là chi phí xây dựng
  - ✦ 40% là chi phí kiểm thử

⌘ Đối với những PM làm theo yêu cầu của khách hàng, chi phí mở rộng thường vượt quá chi phí xây dựng

57/66

### Bản chất của chi phí CNPM

⌘ Chi phí biến đổi tùy thuộc vào :

- Từng loại hệ thống được xây dựng
- Các yêu cầu về đặc điểm của hệ thống :
  - ✦ Hiệu năng của hệ thống
  - ✦ Độ tin cậy của hệ thống

⌘ Việc phân bổ chi phí phụ thuộc vào mô hình phát triển hệ thống được sử dụng

58/66

### Thách thức đối với CNPM

⌘ CNPM trong thế kỷ 21 phải đối mặt với rất nhiều thách thức

⌘ Người ta phải có giải pháp cụ thể với mỗi thách thức

⌘ Nguyên tắc :

- Không đồng nhất :  
Phát triển các kỹ thuật xây dựng PM để giải quyết sự không đồng nhất về môi trường thực hiện và nền tảng hạ tầng
- Chuyển giao :  
Phát triển các kỹ thuật nhằm dẫn tới việc chuyển giao PM tới NSD nhanh hơn
- Độ tin cậy :  
Phát triển các kỹ thuật để chứng minh rằng PM được NSD tin tưởng

59/66

### Tính chuyên nghiệp và đúng quy tắc

⌘ Quy trình xây dựng PM :

- được thực hiện trong một môi trường chuyên nghiệp
- đòi hỏi tuân thủ các nguyên tắc một cách chính xác

⌘ Kỹ sư (hay chuyên gia, cán bộ) PM :

- Phải có trách nhiệm trong công việc, không đơn thuần chỉ là ứng dụng kỹ thuật
- Có thái độ ứng xử trung thực
- Có cách làm việc chuyên nghiệp và đúng quy tắc

60/66