



ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
UNIVERSITY OF SCIENCE AND TECHNOLOGY - UD



CÔNG NGHỆ PHẦN MỀM

TS. Võ Đức Hoàng

Khoa Công nghệ Thông tin

Trường Đại học Bách khoa - Đại học Đà Nẵng

THINKING - CREATING - HUMANITY CHERISHING





ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
UNIVERSITY OF SCIENCE AND TECHNOLOGY - UD



CHƯƠNG 1

Tổng quan về Công nghệ phần mềm

THINKING - CREATING - HUMANITY CHERISHING



NỘI DUNG MÔN HỌC

1. Phần mềm là gì?
2. Phân loại phần mềm
3. Công nghệ phần mềm là gì?
4. Các vấn đề trong công nghệ phần mềm

Mục tiêu của bài học

- Hiểu được các khái niệm cơ bản như: **phần mềm** và **công nghệ phần mềm**
- Biết được **các pha trong phát triển phần mềm**
- Biết được **những khó khăn, thách thức** gặp phải trong quá trình phát triển phần mềm

1. Phần mềm là gì?

- **Định nghĩa (*):**
 - **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

Phần mềm bao gồm:

- **Chương trình máy tính**
 - Phần vận hành được: mã máy
 - Không vận hành: mã nguồn
- **Các cấu trúc dữ liệu**
 - Cấu trúc làm việc (bộ nhớ trong)
 - Cấu trúc lưu trữ
- **Tài liệu sử dụng**
 - Hướng dẫn sử dụng
 - Tham khảo kỹ thuật
 - Tài liệu đặc tả, phân tích, thiết kế, kiểm thử

1. Phần mềm là gì?

- **Các đặc trưng của phần mềm:**
 - Là hàng hóa **vô hình**, không nhìn thấy được
 - Chất lượng phần mềm: không mòn đi mà có **xu hướng tốt lên** sau mỗi lần có lỗi (error) được phát hiện và sửa
 - Phần mềm vốn **chứa lỗi tiềm tàng**, theo quy mô càng lớn thì khả năng chứa lỗi càng cao
 - **Lỗi phần mềm** dễ được phát hiện bởi người ngoài
 - Chức năng của phần mềm **thường biến hóa**, thay đổi theo thời gian (theo nơi sử dụng)

2. Phân loại phần mềm

- Phần mềm **hệ thống** (System SW)
- Phần mềm **thời gian thực** (Real-time SW)
- Phần mềm **ng nghiệp vụ** (Business SW)
- Phần mềm **KH&KT** (Engineering & Science SW)
- Phần mềm **nhúng** (Embedded SW)
- Phần mềm **máy cá nhân** (Personal computer SW)
- Phần mềm **trên Web** (Web-based SW)
- Phần mềm **trí tuệ nhân tạo** (Artificial Intelligent SW)

Phân biệt
**SOFTWARE và
PROGRAMS**

Phần mềm - Chương trình - Ứng dụng?

- **Phần mềm - Software là gì**

- Phần mềm máy tính là tập hợp các dữ liệu và các chương trình điều khiển hoạt động của máy vi tính. Thuật ngữ phần mềm - Software được dùng để phân biệt với phần cứng - Hardware là các bộ phận vật lý của máy vi tính.

- **Chương trình - Program là gì**

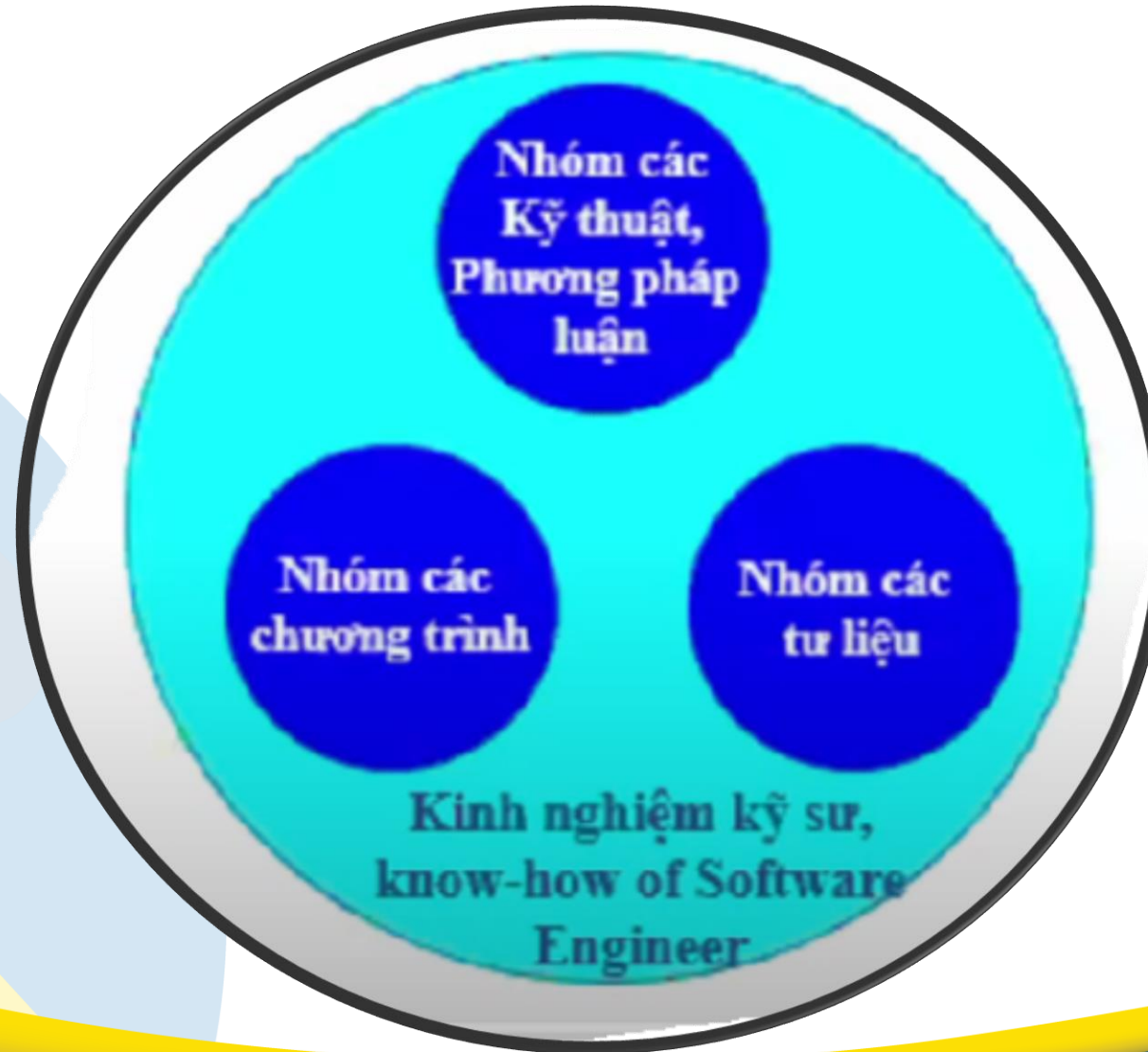
- Chương trình là một tập hợp các lệnh các hướng dẫn cho máy vi tính xử lý thông tin đầu vào sau đó thao tác với dữ liệu và đưa ra kết quả. Thông thường, chương trình được đưa vào một vùng lưu trữ mà máy tính có thể truy cập được. Máy tính nhận một lệnh và thực hiện nó và sau đó nhận lệnh tiếp theo.

- **Ứng dụng - Application là gì**

- Ứng dụng là một chương trình hoặc một nhóm chương trình được thiết kế cho người dùng cuối thực hiện một công việc chung hoặc cụ thể nào đó.

- **Phân biệt các khái niệm sau:**
 - **Hệ thống, phần mềm, ứng dụng**
 - **Lập trình, phát triển phần mềm**
 - **Lập trình viên và kỹ sư phần mềm**

Định nghĩa phần mềm



3. Công nghệ phần mềm (Software Engineering)

▪ Định nghĩa

- Bauer [1969]: CNPM là **việc thiết lập và sử dụng** các **nguyên tắc công nghệ** học đúng đắn dùng để thu được phần mềm một cách **kinh tế** vừa **tin cậy** vừa làm việc **hiệu quả** trên các máy thực
- Parnas [1987]: CNPM là việc xây dựng phần mềm nhiều phiên bản bởi nhiều người
- Ghezzi [1991]: CNPM là một lĩnh vực của khoa học máy tính, liên quan đến xây dựng các hệ thống phần mềm vừa lớn vừa phức tạp bởi một hay một số nhóm kỹ sư.

3. Công nghệ phần mềm (Software Engineering)

- **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 dùng trong (1)
- **Pressman [1995]: CNPM là bộ môn tích hợp cả quy trình, các phương pháp, các công cụ để phát triển phần mềm máy tính**

3. Công nghệ phần mềm (Software Engineering)

- **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**

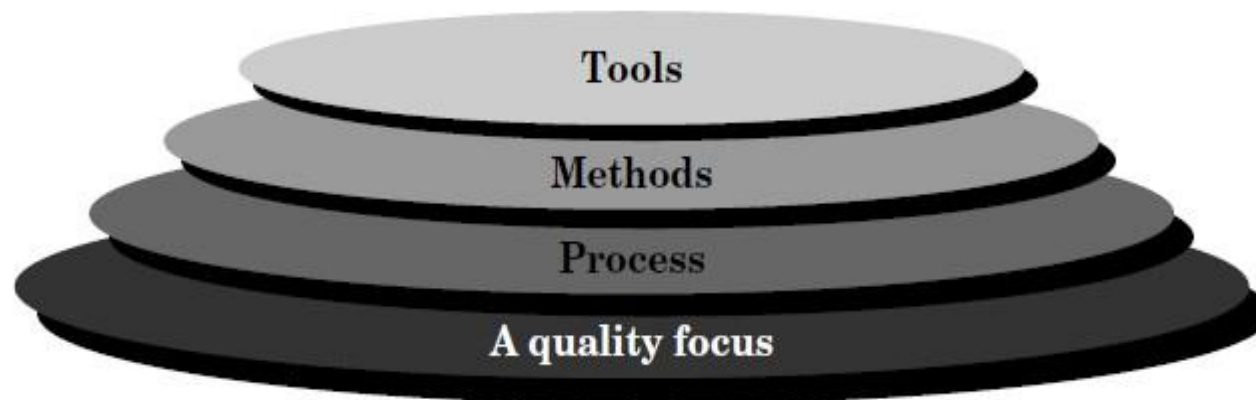
- Công nghệ phần mềm là lĩnh vực **khoa học** về các **phương pháp luận**, **kỹ thuật** và **công cụ** tích hợp trong quy trình sản xuất và vận hành phần mềm nhằm **tạo ra phần mềm** với những chất lượng mong muốn

[Software Engineering is a scientific field to deal with methodologies, techniques and tools integrated in software production-maintenance process to obtain software with desired qualities]

Các mục tiêu chính

- 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)
- Các phương pháp (Methods)
- Các công cụ (Tools)

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

- 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

- Được phân thành ba giai đoạn chung
 - Pha **định nghĩa** (Definition phase)
 - Pha **phát triển** (Development phase)
 - Pha **hỗ trợ** (Support phase)

- 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.

- 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.

- 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).**



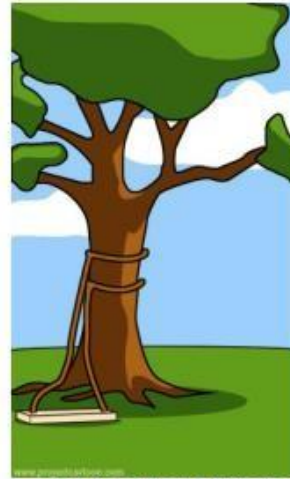
How the customer explained it



How the project leader understood it



How the analyst designed it



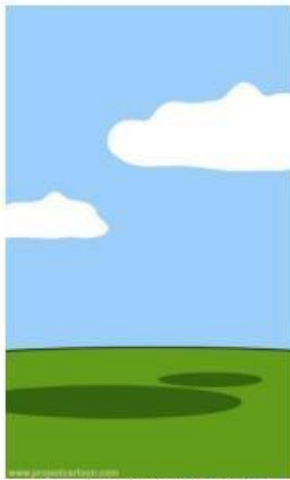
How the programmer wrote it



What the beta testers received



How the business consultant described it



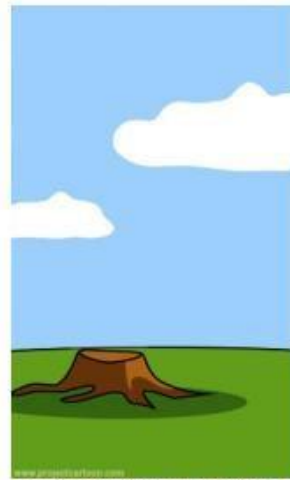
How the project was documented



What operations installed



How the customer was billed



How it was supported



What marketing advertised



What the customer really needed

4. Các vấn đề

- **Nhu cầu kinh doanh không được đáp ứng**
- **Yêu cầu không được giải quyết**
- **Các module không tích hợp**
- **Khó khăn khi bảo trì**
- **Phát hiện muộn về sai sót**
- **Chất lượng trải nghiệm kém**
- **Hiệu suất kém**
- **Không có nỗ lực phối hợp của nhóm**
- **Các vấn đề về xây dựng và phát hành**

4. Các vấn đề

- *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*
- *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*

4. Các vấn đề

- *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*

4. Các vấn đề

- 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

4. Các vấn đề

- 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

Tổng kết



- Phần mềm gồm: *chương trình, dữ liệu, tài liệu*
- Công nghệ phần mềm:
 - *khoa học* về các *phương pháp luận, kỹ thuật* và *công cụ* trong quy trình sản xuất và vận hành phần mềm nhằm *tạo ra phần mềm* với những chất lượng mong muốn
 - Các pha: *định nghĩa, phát triển, hỗ trợ*
- Một số khó khăn:
 - Không đáp ứng được nhu cầu
 - Khó khăn khi bảo trì
 - Phát hiện muộn về sai sót

