**Nội dung ôn thi Công nghệ phần mềm**

- Ôn lý thuyết công nghệ phần mềm chương 1 , 2 , 3

- Vẽ sơ đồ chương 4, 5

**Kiến thức ôn thi Công nghệ phần mềm**

**Mục lục**

**Chương 1: Giới thiệu CNPM 2**

**Câu 1: Lịch sử phát triển của 2**

**Công nghệ phần mềm**

**Câu 2: Định nghĩa và mục tiêu của 2**

**công nghệ phần mềm**

**Câu 3: Chất lượng bên ngoài và các 3**

**tiêu chí đánh giá bên ngoài**

**Câu 4: Chất lượng bên trong và các 3**

**tiêu chí đánh giá bên trong**

**Câu 5: Các loại phần mềm 5**

**CHƯƠNG 2: QUẢN LÝ ĐỀ ÁN PHẦN MỀM 5**

**Câu 1: Hãy nêu các tiến trình phổ biến hiện nay 5**

**Câu 2: Trình bày mô hình thác nước (Waterfall Model) 6**

**Vẽ hình và mô tả các giai đoạn trong quy trình thác**

**Câu 3: Trình bày mô hình (ProtoType). Vẽ hình 7**

**và mô tả các quy trình trong ProtoType**

**CHƯƠNG 3: QUẢN LÝ ĐỀ ÁN PHẦN MỀM 8**

**Câu 1: Subteam trong CNPM là gì? 8**

**Câu 2: hãy trình bày các subteam 9**

**Chương 1: Giới thiệu Công nghệ phần mềm**

**Câu 1: Lịch sử phát triển của Công nghệ phần mềm**

- 1968: Hội nghị khoa học của NATO đã đưa ra từ “Software Engineering” (Công nghệ phần mềm). Bắt đầu bàn luận về khủng hoảng phần mềm (Software Crisis) và xu hướng hình thành CNPM như 1 chuyên môn riêng.

- Nửa cuối 1968: IBM đưa ra chính sách phân biệt giá cả giữa phần cứng và phần mềm. Từ đó, ý thức về phần mềm ngày càng cao. Bắt đầu những nghiên cứu cơ bản về phương pháp luận lập trình.

- Nửa đầu những năm 1970: Nhằm nâng cao chất lượng phần mềm, không chỉ có các nghiên cứu về lập trình, kiểm thử, mà còn có cả những nghiên cứu đảm bảo tính tin cậy trong quá trình sản xuất phần mềm.

- Năm 1975: Hội nghị quốc tế đầu tiên về CNPM được tổ chức: International Conference on SE (ICSE).

- Nửa sau những năm 1970: Quan tâm đến mọi pha trong quá trình phát triển phần mềm, nhưng tập trung chính ở những pha đầu.

- Nửa đầu những năm 1980: Trình độ học vấn và ứng dụng CNPM được nâng cao, các công nghệ được chuyển vào thực tế. Xuất hiện các sản phẩm phần mềm và các công cụ khác nhau làm tăng năng suất sản xuất phần mềm đáng kể.

- Nửa cuối những năm 1980 đến nay: Từ học vấn sang nghiệp vụ. Chất lượng phần mềm tập trung chủ yếu ở tính năng suất, độ tin cậy và tính bảo trì. Nghiên cứu tự động hóa sản xuất phần mềm.

**Câu 2: Định nghĩa và mục tiêu của công nghệ phần mềm**

• Định nghĩa CNPM: Theo Roger S. Pressman - CNPM là bộ môn tích hợp cả các quy trình, các phương pháp, các công cụ để phát triển phần mềm máy tính.

• Mục tiêu:

• Là cung cấp 1 cấu trúc cho việc xây dựng phần mềm có chất lượng cao: tính đúng đắn và độ tin cậy cao, dễ sử dụng, thân thiện với người dùng, dễ hiểu.

**Câu 3: Chất lượng bên ngoài và các tiêu chí đánh giá bên ngoài**

• Những yếu tố này sẽ bao gồm cả:

• Thuộc tính chức năng (functional attributes).

• Thuộc tính phi chức năng (non-functional attributes).

• Những thuộc tính chức năng sẽ miêu tả những chức năng mà sản phẩm phần mềm phải thực hiện (describe WHAT the product MUST do)

• Những thuộc tính phi chức năng lại miêu tả về cách thức chương trình thực thi (describe HOW the product SHOULD be implemented).

Các tiêu chí đánh giá bên ngoài

• Tính dễ sử dụng (usability): giao diện có thân thiện không? Các thao tác thực hiện có gần gũi không? v.v

• Tính tin cậy (reliability): các chức năng của chương trình đều thực hiện đúng chứ? Các công thức tính toán đều cho ra kết quả đúng như mong muốn? các dữ liệu được lưu vào trong DB đúng như mong muốn? phần mềm chạy ổn định?

• Tính chức năng (functionality): từng chức năng đều thực hiện đúng? Các công thức tính toán đều cho ra kết quả đúng như mong muốn? Các dữ liệu được lưu vào trong DB đúng như mong muốn?...

• Tính bền vững (stability): phần mềm có thể hoạt động trong những điều kiện khác nhau? Trong những môi trường khác nhau?

• Tính tương thích (compatibility): phần mềm có thể dễ dàng tích hợp với các sản phẩm phần mềm khác?

• Tinh thực thi (performance): phần mềm chạy với tốc độ nhanh hay chậm? khi chạy có sử dụng nhiều tài nguyên của máy tính không: bộ nhớ, bộ xử lý,...?

**Câu 4: Chất lượng bên trong và các tiêu chí đánh giá bên trong**

• Những yếu tố chất lượng bên trong là những yếu tố “trong suốt” với người dùng, chỉ những người phát triển (developer) mới thấy được.

• Những yếu tố này là những tài liệu tham gia vào quá trình phát triển phần mềm, như: tài liệu phân tích yêu cầu, tài liệu thiết kế,...và đoạn code.

Các tiêu chí đánh giá bên trong

### **1. Tính đúng**

Một sản phẩm thực hiện được gọi là đúng nếu nó thực hiện chính xác những chức năng đã đặc tả và thỏa mãn các mục đích công việc của khách hàng.

### **2. Tính kiểm thử được**

Phần mềm có thể kiểm thử được là phần mềm mà nó có cách dễ dàng để có thể kiểm tra được. Đảm bảo rằng nó thực hiện đúng các chức năng dự định.

### **3. Tính an toàn**

Tính an toàn của sản phẩm phần mềm được đánh giá thông qua:

* Có cơ chế bảo mật và bảo vệ các đối tượng do hệ thống phát sinh hoặc quản lý.
* Bản thân sản phẩm được đặt trong một cơ chế bảo mật nhằm chống sao chép trộm hoặc làm biến dạng sản phẩm đó.

### **4. Tính toàn vẹn**

Sản phẩm phần mềm có tính toàn vẹn khi nó:

* Có cơ chế ngăn ngừa việc thâm nhập bất hợp pháp vào phần mềm hay dữ liệu và ngăn ngừa việc phát sinh ra những đối tượng (dữ liệu, đơn thể...) sai quy cách hoặc mâu thuẩn với các đối tượng sẳn có.
* Không gây ra nhập nhằng trong thao tác. Đảm bảo nhất quán về cú pháp.
* Có cơ chế phục hồi lại toàn bộ hoặc một phần những đối tượng thuộc toàn bộ hoặc một phần những đối tượng thuộc diện quản lý của sản phẩm trong trường hợp có sự cố như hỏng máy, mất điện đột ngột.

### **5. Tính tiêu chuẩn và tính chuẩn**

Sản phẩm phần mềm cần đạt được một số tiêu chuẩn tối thiểu được thừa nhận trong thị trường hoặc trong khoa học, và có thể chuyển đổi dạng cấu trúc dữ liệu riêng của hệ thống sang chuẩn và ngược lại.

### **6. Tính độc lập**

Phần mềm cần và nên đảm bảo được tính độc lập với các đối tượng sau:

* Độc lập với thiết bị,
* Độc lập với cấu trúc của đối tượng mà sản phẩm đó quản lý,
* Độc lập với nội dung của đối tượng mà sản phẩm đó quản lý.

**Câu 5: Các loại phần mềm**

• System socware (phần mềm hệ thống)

• Real-time software (phần mềm thời gian thực)

• Business socware (phần mềm nghiệp vụ)

• Engineering and scientific software (phần mềm khoa học và ứng dụng) • Embedded socware (phần mềm nhúng)

• Personal computer software (phần mềm máy tính cá nhân)

• Web-based software (phần mềm ứng dụng web)

• Artificial Intelligence software (phần mềm trí tuệ nhân tạo)

**CHƯƠNG 2: QUẢN LÝ ĐỀ ÁN PHẦN MỀM**

**Câu 1: Hãy nêu các tiến trình phổ biến hiện nay**

• Mô hình thác nước (the waterfall model)

• Mô hình mẫu (the prototyping model)

• Mô hình phát triển ứng dụng nhanh RAD (the rapid application development model)

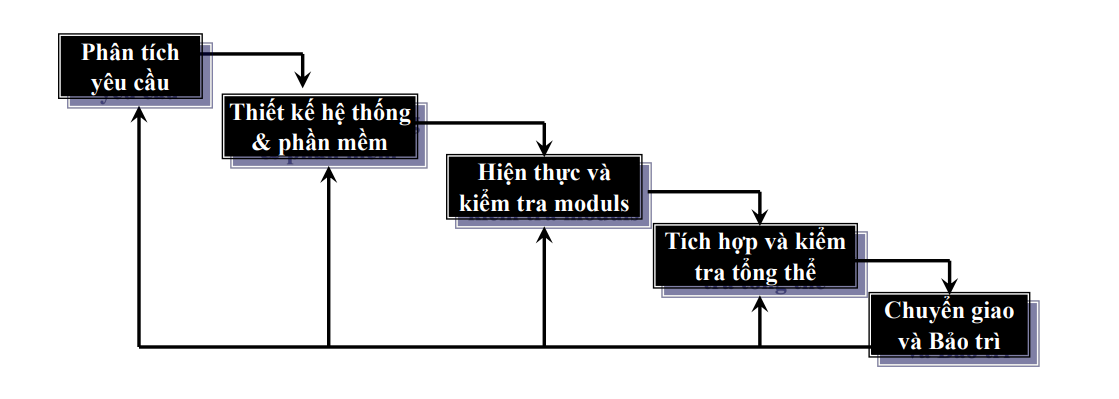
• Mô hình tiến hóa (evolutionary development model)

+ Mô hình gia tăng (incremental model)

+ Mô hình xoắn ốc (the spiral model)

+ Mô hình xoắn ốc WINWIN (the WINWIN spiral model)

• Mô hình Agile

**Câu 2: Trình bày mô hình thác nước (Waterfall Model). Vẽ hình và mô tả các giai đoạn trong quy trình thác nước.**

• Bản chất của phát triển phần mềm là quá trình lặp đi lặp lại chứ không phải tuần tự

• Các bước thực chất không tách biệt hoàn toàn mà có chồng lấn và tham khảo lại

• Bắt buộc khách hàng đặc tả tất cả yêu cầu một cách chính xác và đầy đủ ngay từ ban đầu

• Khách hàng thường phải chờ đợi rất lâu để thấy được phiên bản đầu Dên của sản phẩm

• Tồn tại “delay” Rch lũy trong nhóm làm việc à dự án thường bị trể.

• Chỉ phù hợp cho dự án nhỏ, đơn giản.

**Phân tích các giai đoạn**

**- Giai đoạn phân tích yêu cầu:**

Trong giai đoạn này, doanh nghiệp phải thu thập các yêu cầu, thấu hiểu mong muốn thực sự của đối tượng mục tiêu. Sau đó, đội ngũ chuyên môn tiến hành thảo luận, phân tích tính khả thi, chuẩn bị tài liệu và cơ sở hạ tầng kỹ thuật. Những hoạt động trên giúp doanh nghiệp xác định hướng đi cho quá trình phát triển trong tương lai.

**- Giai đoạn thiết kế hệ thống và phần mềm**

Từ những yêu cầu được xác định trong phân tích yêu cầu, nhóm dự án tạo ra thiết kế cho sản phẩm để đáp ứng tất cả các yêu cầu đó, bao gồm cả thiết kế phần cứng, thiết kế phần mềm, ngôn ngữ lập trình, lưu trữ dữ liệu. Đây đồng thời cũng là phần giúp bạn xác định dự án sẽ hữu ích thế nào đối với người dùng. Nếu bước này gặp vấn đề thì rất có thể phải quay lại bước 1 để thực hiện lại.

**- Giai đoạn hiện thực và kiểm tra moduls**

Khi hệ thống đã được thiết kế đầy đủ và cụ thể, các module chức năng của sản phẩm sẽ được thực hiện trong giai đoạn này để đáp ứng các tiêu chuẩn đã thực hiện ở bước trước. Đây là giai đoạn mà các nhiệm vụ công việc được thảo luận ở bước 2 được tiến hành và cũng là giai đoạn mà đội ngũ lập trình sẽ là nguồn lực chủ yếu được sử dụng.

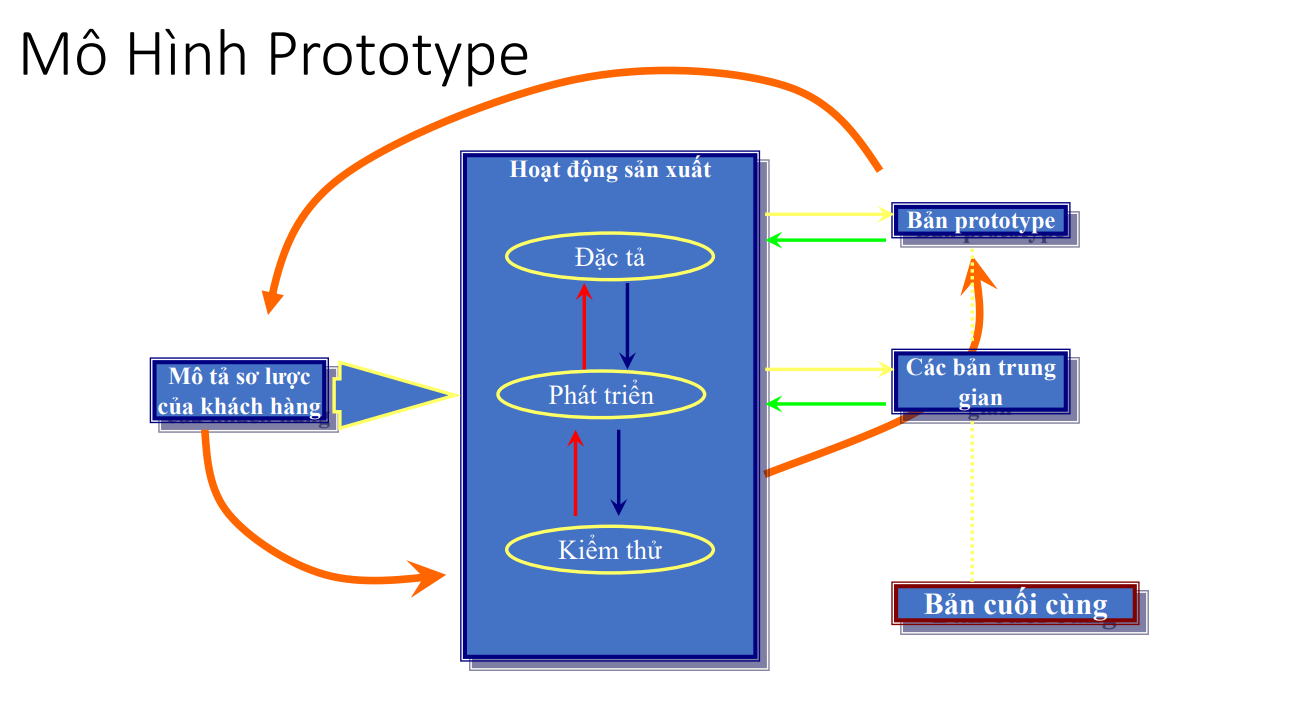
**- Giai đoạn tích hợp và kiểm thử tổng thể**

Ở giai đoạn này, thường sẽ là công việc của đội ngũ QA và tester nhằm tìm kiếm và báo cáo các lỗi trong hệ thống cần được xử lý. Việc này bao gồm tất cả các hoạt động kiểm thử tính năng và phi tính năng. Đây là giai đoạn cực kỳ quan trọng mà nhóm không được phép mắc sai lầm nhằm đảm bảo hệ thống được kiểm tra đầy đủ, các mục tiêu thiết kế và chức năng người dùng yêu cầu được đáp ứng và các nhu cầu kinh doanh được giải quyết.

**- Giai đoạn chuyển giao và bảo trì**

Đây là giai đoạn mà sản phẩm được triển khai vào môi trường mà người dùng có thể bắt đầu sử dụng được. Hay nói cách khác là giai đoạn mà sản phẩm thực sự đi vào hoạt động. Trong giai đoạn này, nhóm dự án cần đảm bảo các yếu tố như: môi trường đang hoạt động, không có lỗi trên server, các tiêu chí test đã được đáp ứng hoặc kiểm tra lại môi trường sau khi ứng dụng được triển khai để đảm bảo sản phẩm không gặp vấn đề…. Sau đó nhóm giải quyết các vấn đề của khách hàng. Trong các dự án phần mềm, đây là giai đoạn các bản được phát hành cập nhật và sửa lỗi.

**Câu 3: Trình bày mô hình (ProtoType). Vẽ hình ProtoType**



• Prototype như là một cơ chế để nhận diện chính xác yêu cầu của khách hàng

• Bản thân khách hàng chưa hiểu rõ yêu cầu của mình, cũng như các quy trình chưa được xác lập rõ ràng.

• Khách hàng chưa hiểu rõ khả năng hổ trợ của hệ thống máy Rnh

• Kích thích sự thích thú của người dùng với dự án

• Prototype có thể bị “throw-away” à Lãng phí

• Các process không được phân định rõ ràng

• Hệ thống thông thường có cấu trúc lỏng lẻo

• Cần có những kỹ năng đăc biệt trong quản lý và phát triển

• Khách hàng hối thúc nhà phát triển hoàn thành sản phẩm một khi thấy được các prototype đầu tiên

**CHƯƠNG 3: QUẢN LÝ ĐỀ ÁN PHẦN MỀM**

**Câu 1: Subteam trong CNPM là gì?**

• Teamwork là mô hình hiện tại cho hầu hết các dự án phần mềm: – Khả năng chuyên nghiệp hóa cao – Hiệu quả trong quản lý, giao tiếp và điều hành

• Một nhóm dự án phần mềm được tạo ra từ nhiều sub-teams >= 1 – Các sub-team không nhất thiết là một nhóm người mà có thể là 1 người – Các sub-team không nhất thiết tồn tại suốt quá trình của một dự án phần mềm.

**Câu 2: hãy trình bày các subteam**

**System Analysis**

Xác định tính khả thi của dự án

• Phân tích chi phí (Cost analysis)

• Dự đoán lợi nhuận (Estimate revenues)

• Tiên liệu các khó khăn về kỹ thuật và công nghệ

•Sau khi nghiên cứu khả thi, nhóm này sẽ làm việc với Requirement Team để nhận feedbacks

•Nếu dự án được phát triển theo mô hình tương tác cao như Prototype/Spiral model thì tính tương tác và feedback là rất quan trọng kể cả với các nhóm khác.

**Planning Team**

Nhóm này có nhiệm vụ xây dựng tổng thể tất cả các kế hoạch quản trị dự án và bảo đảm các tiến trình diển ra đúng tiến độ đã định

•Xây dựng các kế hoạch thực hiện

•Lập các time frame cho các tiến trình

•Kế hoạch sử dụng tài nguyên của hệ thống bao gồm cả nhân lực

•Các kế hoạch dự phòng và điều chỉnh khi có sự cố

**Requirement Team**

Tiếp xúc khách hàng và xác định đầy đủ, hoàn chỉnh và chính xác các yêu cầu cho dự án

•Dùng các phương thức gặp gở chính thức và bên lề để xác định các yêu cầu của hệ thống

•Nếu không có khách hàng, có thể tiếp xúc với các user tiềm năng Sau khi xác định các yêu cầu, nhóm này sẽ làm việc với System Design Team để nhận các feedback. Nếu dự án được phát triển theo mô hình tương tác cao như Prototype/Spiral model thì tính tương tác và feedback là rất quan trọng kể cả với các nhóm khác

**System Design Team**

•Xây dựng thiết kế chi tiết của hệ thống sau khi các yêu cầu đã được xác định.

•Nếu sử dụng mô hình Waterfall, nhóm này phải feedback cho nhóm Requirement những khó khăn nếu có.

•Sau khi hoàn chỉnh thiết kế, nhóm này phải cộng tác với Implementation Team để nhận feedback.

•Nếu dự án được phát triển theo mô hình tương tác cao như Prototype/Spiral model thì tính tương tác và feedback là rất quan trọng kể cả với các nhóm khác

Implementation Team Phát triển hệ thống theo thiết kế đã có.

• Coding

• Kiểm tra cấp Module Sau khi hoàn tất chương trình, nhóm này sẽ cộng tác với nhóm Tesing & Integration để kiểm tra các module Nếu dự án được phát triển theo mô hình tương tác cao như Prototype/Spiral model thì tính tương tác và feedback là rất quan trọng kể cả với các nhóm khác

**Testing & Integration Team**

•Xây dựng thiết kế chi tiết của hệ thống sau khi các yêu cầu đã được xác định.

•Nếu sử dụng mô hình Waterfall, nhóm này phải feedback cho nhóm Requirement những khó khăn nếu có.

•Sau khi hoàn chỉnh thiết kế, nhóm này phải cộng tác với Implementation Team để nhận feedback.

•Nhóm này có thể tiếp nhậncác module rời rạc và kiểm tra sau đó tích hợp thành hệ thống hoàn chỉnh.

•Nếu dự án được phát triển theo mô hình tương tác cao như Prototype/Spiral model thì tính tương tác và feedback là rất quan trọng kể cả với các nhóm khác

•Nhóm này cũng có vai trò trong Interface Control Document để đặc tả các giao diện và giao tiếp giữa các thành phần trong hệ thống

**Trainning Team**

Chuẩn bị các công cụ và tài liệu cho việc trainning cho người dùng

•Kế hoạch trainning

•Các tài liệu giảng dạy

**Delivery & Installation Team**

Nhiệm vụ là cài đặt hệ thống cho khách hàng và các hỗ trợ kỹ thuật trong cài đặt vận hành hệ thống.

**Maintenance Team**

Bảo trì hệ thống sau khi chuyển giao và cài đặt

•Cập nhật sửa chữa

•Nâng cấp mở rộng Cộng tác chặt chẻ với nhóm implementation để thực hiện việc maintenance

**Quality Assurance Team**

Nhóm này có 2 nhiệm vụ 1. Thiết lập các tiêu chuẩn cho các quá trình sản xuất cũng như tiêu chuẩn thực hiện của sản phẩm phần mềm 2. Cung cấp các cơ chế kiểm tra, kiểm soát nhằm đánh giá khả năng thỏa mãn các tiêu chuẩn tương ứng của các nhóm làm việc.

• Các tiêu chuẩn này dùng trong nội bộ và không chia sẻ với khách hàng. • Các tiêu chuẩn có thể được công bố khi cần thiết, vì vậy cần được lưu trữ và báo cáo cho project manager để hoạt động với bộ phận Q&A

**Metrics Team**

Lưu trữ các thông tin thống kê về các hoạt động của các TEAM trong dự án.

•Số lượng các yêu cầu maintenance

•Số lượng thực hiện dịch vụ maintenance

•Số dòng code được viết

•Thời gian thực hiện từng công việc Nhóm này làm việc với hầu hết các nhóm để cung cấp báo cáo về chất lượng, hiệu quả, đồng thời feedback cho các nhóm đó về hiệu quả công việc.

**Documentation Team**

Nhóm này thực hiện các hoạt động thiết lập các tài liệu cho hệ thống

• Tài liệu về phân tích, thiết kế, hiện thực, source code,..

• Tài liệu hổ trợ : userguide, manual, support document

**System Administrationm Team**

•Nhóm này có nhiệm vụ cung cấp và bảo đảm các hoạt động của các hệ thống hạ tầng kỹ thuật cần thiết cho dự án

•Nhóm này thông thường bao gồm cả Network Administration Team

**Reuse & Reengineering Team**

• Chọn lựa và quyết định việc tái sử dụng các module đã có

• Việc tái sử dụng công nghệ cũng cần thiết khi mà việc phát triển đòi hỏi phải dùng đến các code cũ khi công nghệ đã thay đổi.