**ĐỀ CƯƠNG ÔN TẬP CNMP**

**Chương 1:**

1. Định nghĩa phần mềm:

- 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.

- Phần mềm là một hệ thống các chương trình có thể thực hiện trên máy tính nhằm hỗ trợ các nhà chuyên môn trong từng lĩnh vực chuyên ngành thực hiện tốt nhất các thao tác nghiệp vụ của mình. Nhiệm vụ chính yếu của phần mềm là cho phép các nhà chuyên môn thực hiện các công việc của họ trên máy tính dễ dàng và nhanh chóng hơn so với khi thực hiện cùng công việc đó trong thế giới thực.

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.

1. Tiêu chí đánh giá trong ngoài của phần mềm
   1. Tiêu chí 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).

• *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,...?

* 1. Tiêu chí trong

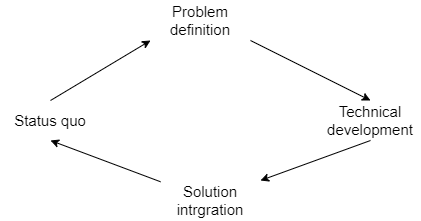
• Những yếu tố chất lượng bên trong là những yếu tố “trong suốt” với người dùng, chỉ những người phát triển (developer) mới thấy được.  
• Những yếu tố này là những tài liệu tham gia vào quá trinh phát triển phần mềm, như: tài liệu phân tích yêu cầu, tài liệu thiết kế,...và đoạn code.

**Chương 2:**

1. Các công việc trong một quy trình phần mềm

• Phân tích hệ thống/vấn đề  
• Xác định các yêu cầu  
• Thiết kế phần mềm  
• Viết phần mềm (coding)  
• Kiểm tra và Rch hợp hệ thống  
• Cài đặt và chuyển giao phần  
mềm  
• Lập tài liệu  
• Bảo trì  
• Quản lý chất lượng  
• Huấn luyện  
• Dự đoán tài nguyên  
• Quản trị dự án

1. Bốn giai đoạn cơ bản của một quy trình

-Đặc tả phần mềm (Software/problem specification): chức năng của phần mềm và những ràng buộc trong họat động của nó phải được định nghĩa.  
-Thiết kế và thực thi phần mềm (Technical development): phần mềm đáp ứng đặc tả phải được tạo ra.  
-Tiến triển phần mềm (Solution Integration): phần mềm phải tiến triển để đáp ứng những nhu cầu thay đổi của khách hàng.  
-Chứng nhận phần mềm (Status Quo): phần mềm phải được chứng nhận để đảm bảo rằng nó làm những gì mà khách hàng muốn.

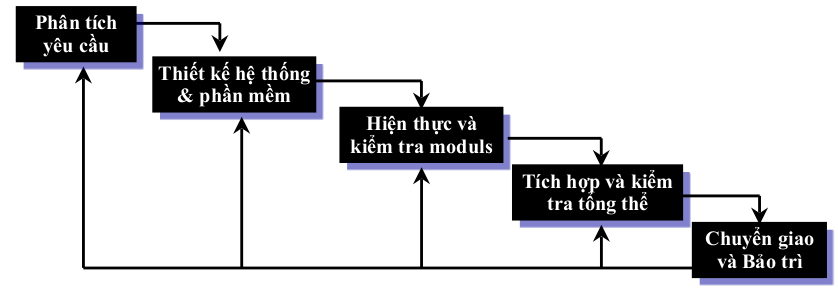
3. Các quy trình

**Quy trình thác nước**

- Mô hình phát triển phần mềm đầu tiên

- Các công việc nối nhau một cách tuần tự

- Đặt nền móng cho các phương pháp phân tích, thiết kế, kiểm tra...



• 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

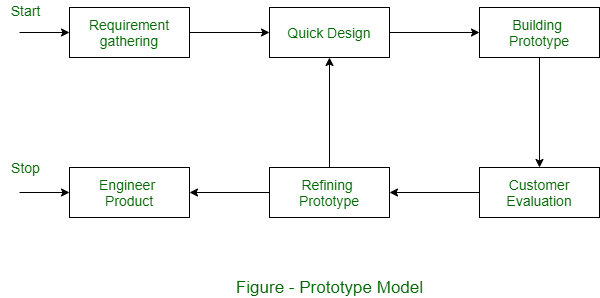
• Bắt buộc khách hàng đặc tả tất cả yêu cầu một cách chính xác và đầy đủ ngay từ ban đầu

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

• Tồn tại “delay” tích lũy trong nhóm làm việc bị trể.

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

**Mô 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 tính

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

• Prototype có thể bị “throw-away” 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

Mô hình prototype - ứng dụng

• Dùng cho các hệ thống nhỏ. Các chi phí khi thay đổi hệ thống là không quá lớn khi cần phải thay đổi sau khi thực hiện prototype

• Cần sự cấp bách về thời gian triển khai ngắn. Hệ thống cần được đưa vào ứng dụng từng phần trong khoảng thời gian nhất định.

• Trong trường hợp những hệ thống mà việc đặc tả các yêu cầu là rất khó và không rõ ràng ngay từ đầu.

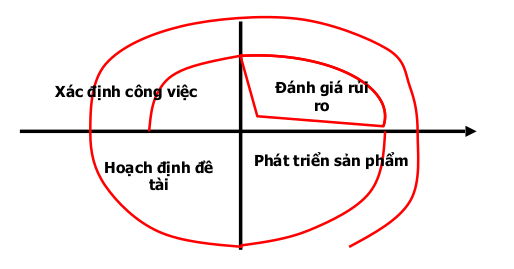
**Mô hìn xoắn ốc**

• Được thực hiện theo một chuỗi lặp kiểu xoắn ốc, mỗi lần lặp cải thiện sản phẩm

• Có phương pháp đánh giá rủi ro

• Có thể áp dụng prototype

• Mỗi lần lặp được cải thiện cho thích nghi với bản chất của đề án



Các giai đoạn:

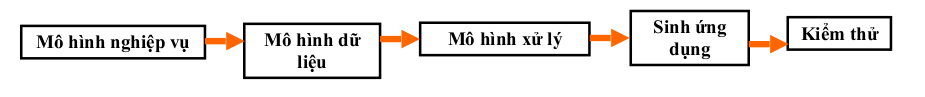
### **Lập kế hoạch - Planning phase:** Thu thập, phân tích yêu cầu từ của dự án từ phía khách hàng. Bao gồm các công việc: Ước lượng chi phí (estimating cost), lên lịch trình thực hiện dự án (shedule-master), xác định số lượng nhân lực, môi trường làm việc (identifying necessary resources and work environment), tìm hiểu yêu cầu hệ thống (requirements) từ đó đưa ra các tài liệu đặc tả (Bussiness Requirement Specifications và System Requirement specifications) để phục vụ cho việc trao đổi giữa khách hàng và phân tích hệ thống sau này.

### **Phân tích rủi ro - Risk analysis phase:** Một quá trình phân tích sẽ được thực hiện để xác định rủi ro và đưa ra các giải pháp thay thế. Một prototype sẽ được tạo ra ở cuối giai đoạn phân tích rủi ro. Nếu có bất kỳ rủi ro nào được tìm thấy trong quá trình này thì các giải pháp thay thế sẽ được đề xuất và thực hiện.

### **Thực thi kỹ thuật - Engineering phase:** Đây là giai đoạn mà dự án được các dev tiến hành code, các tester tiến hành test và deploying software trên trang web của khách hàng.

### **Đánh giá - Evaluation phase:** Khách hàng sẽ tham gia vào giai đoạn này để đánh giá công việc, sản phẩm và đảm bảo rằng sản phẩm đáp ứng tất cả các yêu cầu đã đặt ra trước đó. Nếu có bất kỳ yêu cầu thay đổi nào từ khách hàng, các giai đoạn sẽ được lặp lại. Đây là giai đoạn quan trọng vì cần có sự phản hồi của khách hàng về sản phẩm trước khi sản phẩm được release.

**Mô hình RAD**

• Rapid Application Development là mô hình tuần tự tuyến tính có thời gian phát triển rất ngắn

• Sử dụng các thành phần có sẵn càng nhiều càng tốt

• Sử dụng công cụ lập trình ở dạng tự động sinh mã chứ không phải các ngôn ngữ truyền thống

• Phụ thuộc vào công nghệ phát triển có tính reusable cao.

• Phát triển hệ thống mẫu

Các giai đoạn:

**Business modeling:** Xác định các luồng thông tin giữa các chức năng khác nhau của công việc.

**Data modeling:** Các thông tin thu thập được từ công đoạn 1 sẽ được sử dụng để định nghĩa ra các đối tượng dữ liệu nó cần thiết cho công việc.

**Process modeling:** Sau khi đã định nghĩa ra các đối tượng dữ liệu thì chúng được chuyển đổi để trở thành các dòng thông tin công việc, tiến đến một vài đối tượng công việc riêng.

**Application generation:** Đây là công đoạn coding giống như trong các quy trình khác, khi dữ liệu và tiến trình thực hiện đã được vạch ra thì công đoạn này là sử dụng các công cụ lập trình, phần trình biên dịch, và các ngôn ngữ lập trình để tiến hành code lên các chức năng phần mềm.

**Step 5. Testing và Turnover.**  
Là công đoạn test sản phẩm được tạo, sau khi việc coding đã hoàn thành, và cuối cùng đưa vào thử nghiệm trong thực tế, giao bán sản phẩm, tính toán doanh thu.

**Chương 3:**

Các kỹ thuật phân tích yêu cầu

* Phỏng vấn

-Chuẩn bị phỏng vấn.

+Đọc trước các tài liệu liên quan

+Xác định mục tiêu của cuộc phỏng vấn.

+Chọn người phỏng vấn (chiến lược chọn mẫu)

+Phỏng vấn cá nhân/nhóm

+Phỏng vấn có tự do/có định hướng

+Thiết kế các câu hỏi, hệ thống hoá các câu hỏi:

\*Câu hỏi đóng

\*Câu hỏi mở

\*Câu hỏi chung chung  
-Thực hiện phỏng vấn.

+Luôn bám sát những gì đã chuẩn bị để làm chủ cuộc phỏng vấn

+Không biến cuộc phỏng vấn thành cuộc hỏi cung, phải tạo không khí thoải mái, có thái độ phù hợp (biết lắng

nghe)

+Biết cách ngắt và tóm tắt lại các nội dung quan trọng để kiểm nghiệm lại

+Quan sát biểu hiện của người được phỏng vấn.  
-Theo dõi sau phỏng vấn.

* Bảng câu hỏi

-Chọn mẫu những người sẽ trả lời bảng câu hỏi.  
-Thông tin trả lời trên bảng câu hỏi cần bảo mật?  
-Thiết kế bảng câu hỏi:

+Câu hỏi trình bày rõ ràng

+Hình thức bảng câu hỏi phải dễ dàng để xử lý tự động  
-Tổng hợp và phân tích các câu trả lời  
-Thông báo kết quả phân tích cho những người tham gia trả lời.

+Nguyên tắc khi lập bảng câu hỏi:

+Bắt đầu bằng câu hỏi quan trọng.

+Gom nhóm những câu hỏi có cùng chủ đề một cách logic.

+Không nên trình bày quá nhiều trong một trang.

+Tránh viết tắt, tránh dùng những cụm từ/ câu hỏi không rõ nghĩa.

+Thường không yêu cầu người trả lời ghi họ tên.

* Phân tích( nghiên cứu) các tài liệu

-Các tài liệu (có thể tìm hiểu những văn bản chung)  
-Những quy định nội bộ  
-Các báo cáo liên quan  
-Những quy định về quy trình nghiệp vụ: Rất khó có đầy đủ văn bản quy định về quy trình nghiệp vụ  
-Phân tích luồng công việc  
-Mô hình hoá các luồng công việc

* Quan sát thực tế

-Việc quan sát trực tiếp những người thực hiện công việc đó giúp ta có thể kiểm tra lại, đồng thời nắm được những tình huống, những chi tiết đặc biệt mà người quản lý có thể không nhớ hoặc không nắm hết.  
-Có thể quan sát định kỳ nhiều lần, có thể thay đổi về thời điểm quan sát. Các lần quan sát phải có mục đích rõ ràng

* Phân tích thiết kế nhóm (JAD - Joint Application Design)

-Do IBM đề nghị  
-Làm việc tập thể, từ 8-12 người  
 Bao gồm chuyên viên hệ thống.  
 Những người sử dụng tương lai sẽ tham gia nhiều nhất vào hệ thống.  
 Những người có quyền yêu cầu và quyết định về chức năng của hệ thống

Định nghĩa chức năng- phi chức năng

-Chức năng: Mô tả hệ thống sẽ làm gì?

* Lưu trữ, Tra cứu, Tính toán, Kết xuất, …

-Phi chức năng: Không đề cập trực tiếp tới các chức năng cụ thể của hệ thống

* Yêu cầu bảo mật, mã hóa dữ liệu, sao lưu định kỳ, phục hội khi có sự cố,…

Bonus:

Yêu cầu chức năng sẽ chỉ ra một hành vi hoặc một chức năng.

Yêu cầu phi chức năng chỉ ra những quy định về tính chất và ràng buộc cho phần mềm hay hệ thống. Chúng chỉ ra những tiêu chí để đánh giá hoạt động của hệ thống.

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………...…………………………………………………………………………….…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………