

# CÔNG NGHỆ PHẦN MỀM

## *(KỸ NGHỆ PHẦN MỀM)*

(30=15+5+10)

- Giảng lý thuyết: ~15 tiết (LT)
- Giải đáp, kiểm tra: ~5 tiết (KT,GD)
- Bài tập nhóm, báo cáo: ~10 tiết (BT)

-----

- Làm bài thi: 60%
  - Bài tập nhóm: 15%
  - Kiểm tra tại lớp: 15%
  - Tham gia lớp: 5%+ 5%
- (điều kiện cần- **vắng < 5LT**)

# Nội dung môn học

---

- Tổng quan phần mềm và kỹ nghệ phần mềm
  - Vai trò và tầm quan trọng của phần mềm
  - Khó khăn và thách thức đối với phát triển phần mềm
  - Tổng quan về kỹ nghệ phần mềm
- Tiến trình và mô hình tiến trình phần mềm
  - Khái niệm
  - Các mô hình tiến trình tổng quát, cụ thể

# Nội dung môn học (tiếp)

---

- Quản lý dự án phát triển phần mềm
  - Khái niệm, vai trò, mục tiêu của quản lý dự án
  - Các yếu tố quản lý
  - Các hoạt động quản lý (phương pháp, công cụ hỗ trợ)
- Phân tích và đặc tả yêu cầu
  - Vai trò của phân tích và đặc tả yêu cầu
  - Các hoạt động phân tích, đặc tả yêu cầu.
  - Phương pháp, công cụ để phân tích, đặc tả yêu cầu.

# Nội dung môn học (tiếp)

---

- Thiết kế phần mềm
  - Khái niệm, nguyên lý, chất lượng thiết kế phần mềm
  - Thiết kế kiến trúc, thiết kế giao diện
  - Một số phương pháp, công cụ thiết kế
- Lập trình
  - Phong cách lập trình
  - Lập trình tránh lỗi
  - Lập trình hướng hiệu quả

# Nội dung môn học (tiếp)

---

- Kiểm thử
  - Khái niệm, các loại kiểm thử
  - Quy trình, công cụ trợ giúp kiểm thử
  - Viết tài liệu kiểm thử
- Tích hợp, chuyển giao và bảo trì
  - Khái niệm
  - Các nguyên lý, hoạt động tích hợp
  - Các hoạt động chuyển giao, bảo trì
  - Các vấn đề của hoạt động bảo trì

# BÀI TẬP MÔN HỌC- THỰC HIỆN THEO NHÓM

## 1. Tạo nhóm:

- ~7 thành viên; Chọn nhóm trưởng;
- Chọn đề tài: đề tài giải quyết bài toán nào?
- Lập bảng đánh giá khả năng chuyên môn của thành viên

## 2. Tạo dự án (khả thi)

- Khảo sát sơ bộ dự án, viết thành tài liệu dự án;
- Họp nhóm để xác định, ước lượng các công việc dự án;
- Phân công nhiệm vụ cho các thành viên dựa trên chuyên môn
- Đánh giá chi phí (thời gian) của từng công việc/cá nhân
- Xác định chi phí (thời gian) cho toàn dự án
- Lập hồ sơ quản lý để báo cáo dự án khả thi

## 3. Thực hiện dự án để tạo sản phẩm

- Thiết kế dự án → xây dựng hồ sơ thiết kế
- Cài đặt ==→ **SẢN PHẨM PHẦN MỀM**

GHI CHÚ: Trong thời gian thực hiện, nhóm trưởng theo dõi tiến độ và lập bảng theo dõi của từng cá nhân.

## TÀI LIỆU THAM KHẢO

1. Lê Văn Tường Lân, Giáo trình Công nghệ phần mềm, Đại Học Huế, 2006
2. Ngô Trung Việt, Kỹ nghệ phần mềm - bản dịch, Nhà xuất bản Giáo dục, 1999.
3. Đoàn Văn Ban, Phân tích thiết kế và lập trình hướng đối tượng, Trung tâm Khoa học Tự nhiên và Công nghệ Quốc gia, 1996.
4. Lê Đức Trung, Công nghệ phần mềm, Nhà xuất bản Khoa học và Kỹ thuật, 2002.
5. Khoa Công nghệ Thông tin - Đại học Khoa học Tự nhiên Hà Nội, Bài giảng về nhập môn công trình học phần mềm, Hà Nội, 1997.
6. Lê Đức Trung, Những viên ngọc trong kỹ thuật lập trình - bản dịch, Trung tâm Tin học và Điện tử Phương Đông, 1992 .
7. Roger S. Pressman Ph.D, Software engineering a practitioner's - 5<sup>th</sup>, McGraw-Hill book Co.-Singapore, 2001.
8. Yourdon, E. Software reuse, Application development strategies, Vol. 6, No. 12, December 1994, pp.1-16.
9. Sommerville I., Software engineering - 4<sup>th</sup>, Addison Wesley, 1995.
10. Eldon Wig, Industry Software engineering course - VCIT project, Intellitec Consulting Inc., Hà Nội 10/1999.

## CHƯƠNG 1

# TỔNG QUAN VỀ CÔNG NGHỆ PHẦN MỀM

Công nghệ phần mềm -Software Engineering- là các hoạt động bao gồm: phát triển, đưa vào hoạt động, bảo trì, và loại bỏ phần mềm một cách có hệ thống. Các kỹ sư phần mềm sẽ được cung cấp với các kỹ thuật, công cụ cơ bản nhằm phát triển các hệ thống phần mềm.

Như vậy, công nghệ phần mềm là lĩnh vực nghiên cứu của tin học, nhằm đề xuất các nguyên lý, phương pháp, công cụ, cách tiếp cận và phương tiện phục vụ cho việc thiết kế và cài đặt các sản phẩm phần mềm có chất lượng.



## 1.1. Công nghệ phần mềm là gì?

### HW

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

### SW

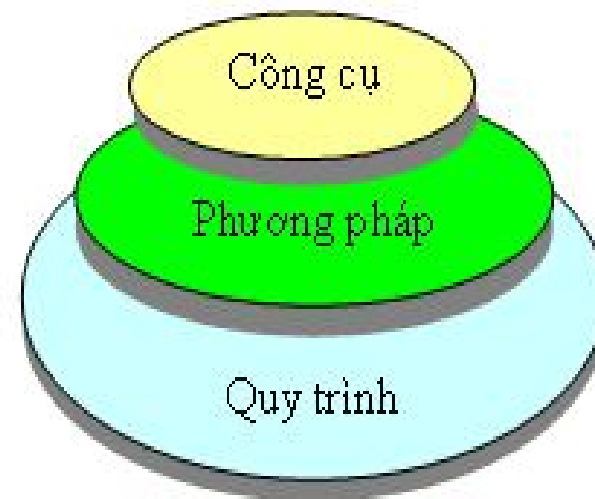
- Vật “mềm”
- Kỹ thuật sử dụng
- Trừu tượng
- Vô hình
- Sản xuất bởi con người là chính
- Định tính là chính
- Không hao mòn

- Phần mềm (Software - SW) như một khái niệm đối nghĩa với phần cứng (Hardware - HW), tuy nhiên, đây là 2 khái niệm tương đối
- Từ xưa, SW như thứ được cho không hoặc bán kèm theo máy (HW)
- Dần dần, giá thành SW ngày càng cao và nay cao hơn HW

Công nghệ phần mềm (software engineering): là việc áp dụng các công cụ, các kỹ thuật một cách hệ thống trong việc phát triển các ứng dụng dựa trên máy tính. Đó chính là việc áp dụng các quan điểm, các tiến trình có kỷ luật và lượng hoá được, có bài bản và hệ thống để phát triển, vận hành và bảo trì phần mềm.

Theo quan điểm của nhiều nhà nghiên cứu, có thể nhìn công nghệ phần mềm là một mô hình được phân theo ba tầng mà tất cả các tầng này đều nhằm tới mục tiêu chất lượng, chi phí, thời hạn phát triển phần mềm.

Mô hình được phân theo ba tầng của công nghệ phần mềm được mô tả như sau:



Ở đây tầng quy trình (process) liên quan tới vấn đề quản trị phát triển phần mềm như lập kế hoạch, quản trị chất lượng, tiến độ, chi phí, mua bán sản phẩm phụ, cấu hình phần mềm, quản trị sự thay đổi, quản trị nhân sự (trong môi trường làm việc nhóm), việc chuyển giao, đào tạo, tài liệu;

Tầng phương pháp (methods) hay cách thức, công nghệ, kỹ thuật để làm phần mềm: liên quan đến tất cả các công đoạn phát triển hệ thống như nghiên cứu yêu cầu, thiết kế, lập trình, kiểm thử và bảo trì. Phương pháp dựa trên những nguyên lý cơ bản nhất cho tất cả các lĩnh vực công nghệ kể cả các hoạt động mô hình hoá và kỹ thuật mô tả.

Tầng công cụ (tools) liên quan đến việc cung cấp các phương tiện hỗ trợ tự động hay bán tự động cho các tầng quá trình và phương pháp (công nghệ).

- Sản xuất phần mềm phụ thuộc rất nhiều vào con người (kỹ sư phần mềm). Khả năng hệ thống hóa trừu tượng, khả năng lập trình, kỹ năng công nghệ, kinh nghiệm làm việc, tầm bao quát, . . . : khác nhau ở từng người
- Phần mềm phụ thuộc nhiều vào ý tưởng (idea) và kỹ năng (know-how) của người/nhóm tác giả

**Kỹ sư phần mềm (software engineer):** là một người biết cách áp dụng rộng rãi những kiến thức về cách phát triển ứng dụng vào việc tổ chức phát triển một cách có hệ thống các ứng dụng. Công việc của người kỹ sư phần mềm là: đánh giá, lựa chọn, sử dụng những cách tiếp cận có tính hệ thống, chuyên biệt, rõ ràng trong việc phát triển, đưa vào ứng dụng, bảo trì, và thay thế phần mềm.

Do đặc điểm nghề nghiệp, người kỹ sư phần mềm phải có những kỹ năng cơ bản như:

- Định danh, đánh giá, cài đặt, lựa chọn một phương pháp luận thích hợp và các công cụ CASE.
- Biết cách sử dụng các mẫu phần mềm (prototyping).
- Biết cách lựa chọn ngôn ngữ, phần cứng, phần mềm.
- Quản lý cấu hình, lập sơ đồ và kiểm soát việc phát triển của các tiến trình.
- Lựa chọn ngôn ngữ máy tính và phát triển chương trình máy tính.
- Đánh giá và quyết định khi nào loại bỏ và nâng cấp các ứng dụng.

Mục tiêu của kỹ sư phần mềm là sản xuất ra các sản phẩm có chất lượng cao và phù hợp với các quy trình phát triển chuẩn mực.

## 1.2. NHÂN TỐ CON NGƯỜI VÀ PHÂN LOẠI NGHỀ NGHIỆP

### 1.3.1. Nhân tố con người trong ngành công nghiệp phần mềm

Đối với một sản phẩm phần mềm, một người không thể hoàn thành mà là kết quả lao động của một nhóm người-ta gọi là nhóm phát triển phần mềm. Mỗi thành viên trong nhóm không được vị kỷ, thành quả lao động của nhóm được xen như là thành quả chung và phải tuyệt đối trung thành với nhóm.

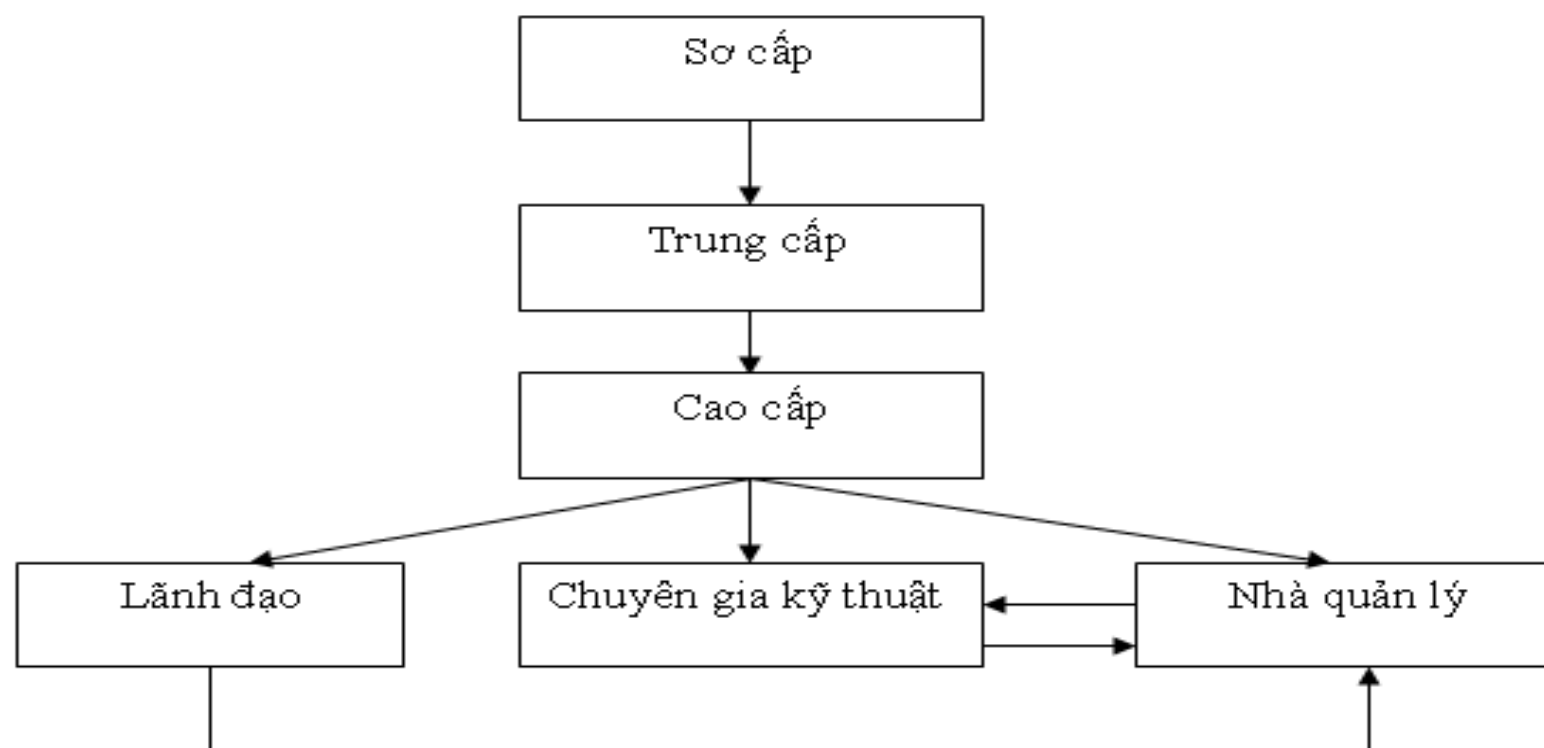
Như vậy, một nhóm phát triển phần mềm như thế nào gọi là một nhóm hợp lý? Sau đây là một vài yếu tố cần xem xét:

- Nhóm có bao nhiêu thành viên,
- Nhóm được tổ chức như thế nào,
- Tình hình thực tế của mỗi thành viên trong nhóm,
- Môi trường, điều kiện mà nhóm đang làm việc,...

Mỗi thành viên trong nhóm phải có một số kiến thức cần thiết tùy thuộc vào vai trò trong nhóm để phát triển phần mềm.

### 1.3.2.1. Mức độ kinh nghiệm

Sơ đồ về mối liên hệ sau được thể hiện như sau:



Mối liên hệ của con đường nghề nghiệp cho các mức khác nhau

### 1.3.2.2. Loại hình công việc

Ở đây, các loại hình công việc được bàn luận đến dựa vào cách phân loại gồm: phát triển ứng dụng, hỗ trợ ứng dụng, chuyên ngành kỹ thuật, nhân viên và những vấn đề khác.

## 1.4. SẢN PHẨM PHẦN MỀM - ĐẶC TÍNH VÀ PHÂN LOẠI

Xây dựng phần mềm là một hoạt động chính của công nghệ phần mềm. Một phần mềm gồm một hay nhiều ứng dụng (application) - là một tập hợp các chương trình thực hiện tự động hóa một số các nhiệm vụ nghiệp vụ. Nghiệp vụ (Business) bao gồm tập hợp các chức năng như: tìm hiểu thị trường, kiểm toán, sản xuất và quản lý nhân sự... Mỗi chức năng có thể được chia nhỏ ra thành những tiến trình thực hiện nó. Các ứng dụng có thể hỗ trợ cho từng nhiệm vụ một cách đơn lẻ. Trái lại một ứng dụng tìm hiểu thị trường có các đặc điểm riêng, có các chức năng riêng, ngoài ra nó còn cung cấp một số thông tin chung nhằm hoàn thiện tất cả các nhiệm vụ.

Mọi ứng dụng đều có một số đặc điểm chung (tương đồng) và một số đặc điểm riêng. Các đặc điểm chung của ứng dụng thường được đề cập là: đặc tính (characteristics), tính đáp ứng (responsiveness) và loại (type) của ứng dụng.



### 1.4.1. Các đặc tính phần mềm

Các đặc tính phần mềm là tất cả các điểm chung cho mọi ứng dụng: dữ liệu, các tiến trình, các ràng buộc, và các giao diện.

#### ***1.4.1.1. Dữ liệu***

Đầu vào: Dữ liệu vào là dữ liệu ở bên ngoài máy tính, và chúng được đưa vào bằng cách sử dụng một thiết bị đầu vào. Thiết bị đầu vào được sử dụng để đưa dữ liệu vào máy tính có thể là: bàn phím, máy quét, hoặc được truyền từ một máy tính khác.

Đầu ra: Dữ liệu ra ngược lại so với dữ liệu vào ở chỗ, đầu ra là các dữ liệu được đưa ra ngoài máy tính. Một số các thiết bị đầu ra như máy in, màn hình hiển thị, một máy tính khác...

Sự lưu trữ dữ liệu và sự tìm kiếm dữ liệu: Dữ liệu được mô tả ở dạng vật lý, trong một máy có thể đọc được các khuôn dạng dữ liệu. Việc tìm kiếm dữ liệu được hiểu là bạn có thể truy nhập vào dữ liệu ở dạng lưu trữ của nó. Việc lưu trữ và tìm kiếm luôn đi cùng với nhau (cả ở mức quan niệm lẫn trong các chương trình phần mềm). Việc lưu trữ dữ liệu đòi hỏi hai kiểu định nghĩa dữ liệu là kiểu vật lý và kiểu logic.

### ***1.4.1.2. Xử lý***

Xử lý bao gồm một chuỗi các lệnh hoặc các sự kiện có liên quan với nhau làm việc với các dữ liệu. Kết quả của một xử lý có thể là: làm thay đổi cơ sở dữ liệu, đưa dữ liệu trả lời ra thiết bị đầu cuối, máy in hoặc in ra giấy, có thể là những yêu cầu về các trang thiết bị, sản sinh những chương trình, hoặc lưu trữ những luật, những thông tin mới, được suy diễn ra về các tình huống, các phần tử.

### ***1.4.1.3. Giao diện***

Quan trọng nhất là giao diện người sử dụng - là phương tiện giao tiếp giữa người sử dụng và chương trình. Sau đó là giao diện thủ công - là các mẫu báo cáo, .... và một số giao diện đã được chuẩn hóa như giao diện về mạng LAN của Institute of Electrical and Electronic Engineers, chuẩn OSI (Open System Interface) của International Standards Organization,...

#### 1.4.1.4. Ràng buộc

Ràng buộc bao gồm: ràng buộc thứ tự trước, ràng buộc thứ tự sau, ràng buộc thời gian, ràng buộc cấu trúc, ràng buộc điều khiển và cả ràng buộc về tham chiếu.

- *Ràng buộc về thứ tự trước (Prerequisite Constraint):* Bắt buộc về thứ tự trước là điều kiện đầu tiên phải được đáp ứng để có thể bắt đầu quá trình xử lý.
- *Ràng buộc về thứ tự sau (Postrequisite Constraint):* Ràng buộc loại này là điều kiện cần phải thỏa mãn để quá trình xử lý có thể hoàn thành được. Cụm câu lệnh này được đưa vào cuối quá trình xử lý.
- *Ràng buộc về thời gian (Time Constraint):* Bao gồm ràng buộc về thời gian xử lý, thời gian phân chia cho một quá trình xử lý, thời gian yêu cầu đối với các quá trình xử lý bên ngoài, thời gian xử lý đồng bộ, thời gian trả lời cho quá trình xử lý với giao diện ngoài.
- *Ràng buộc về mặt cấu trúc:* Có thể hiểu là bao gồm việc xác định loại đầu vào và đầu ra của dữ liệu nào được cho phép, quá trình xử lý được thực hiện như thế nào và mối quan hệ giữa các quá trình với nhau.
- *Ràng buộc về điều khiển:* Liên quan đến việc duy trì mối quan hệ về dữ liệu.
- *Ràng buộc về suy diễn:* Đó là những khả năng có thể xảy ra từ một ứng dụng, dựa vào các kết quả trước đó, hoặc có thể dựa vào các quan hệ về dữ liệu, ta có thể dẫn đến một kết quả khác.

### 1.4.3. Phân loại phần mềm

Phân loại phần mềm được định nghĩa như sự định hướng các công việc của một ứng dụng, ví dụ như theo kiểu hướng giao dịch, hỏi đáp, trợ giúp quyết định,...

#### *1.4.3.1. Ứng dụng hướng giao dịch*

Ứng dụng hướng giao dịch còn có tên là hệ thống xử lý giao dịch (TPS – Transaction Processing Systems) được sử dụng nhằm hỗ trợ các hoạt động hằng ngày của một công việc, bao gồm: xử lý đơn hàng, quản lý kiểm kê, ghi quỹ,... Chúng được đặc trưng như là các ứng dụng mà trong đó các yêu cầu, các dữ liệu và quá trình xử lý được biết rõ và có cấu trúc tốt. Theo nghĩa được biết rõ, chức năng đó phải có tính lặp lại, thân thiện và rõ ràng. Theo nghĩa cấu trúc tốt, vấn đề đó phải có thể được xác định một cách đầy đủ và rõ ràng. Các yêu cầu có thể được định danh bởi đội ngũ xây dựng nhân mềm

### ***1.4.3.2. Ứng dụng cơ sở dữ liệu***

Ứng dụng cơ sở dữ liệu được sử dụng như một ứng dụng xử lý câu hỏi về dữ liệu. Ngôn ngữ truy vấn dữ liệu chuẩn SQL cho phép người sử dụng đặt câu hỏi dưới dạng: họ biết họ cần gì nhưng không biết làm cách nào để lấy được dữ liệu đó. Các phần mềm máy tính đưa ra các phương pháp xử lý và truy cập tối ưu để thực hiện các thao tác đó.

Ở đây, có ba loại câu hỏi chính:

- 1- Tương tác: dữ liệu sử dụng xong là không cần nữa?
- 2- Dữ liệu được lưu trữ để sử dụng lại và thay đổi trong tương lai?
- 3- Dữ liệu được lưu trữ để sử dụng thường xuyên trong một số quá trình lặp lại?

Ứng dụng truy vấn hỗ trợ một khái niệm là kho chứa dữ liệu (data warehouse). Đó là một sơ đồ lưu trữ xây dựng trên quan điểm: hầu hết dữ liệu cần phải giữ lại cho các truy nhập truy vấn trực tuyến. Tại đây lưu lại các phiên bản cũ của phần lớn các phần tử trong cơ sở dữ liệu, các lần vào ra giao dịch và các bản ghi liên quan đến một số quá trình hoạt động.

## 1.5. MỘT SỐ MÔ HÌNH XÂY DỰNG PHẦN MỀM

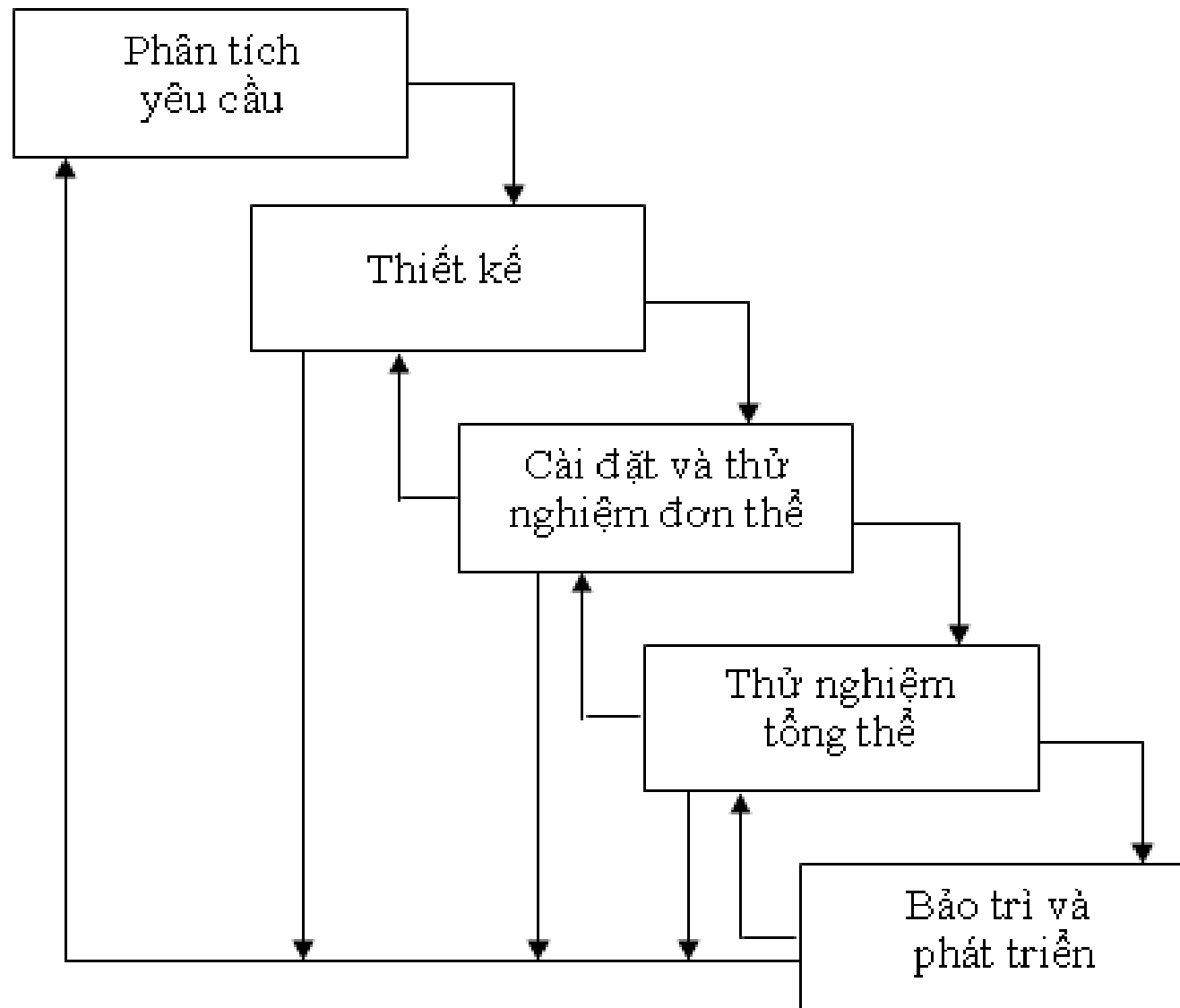
### 1.5.1. Mô hình tuyến tính (The linear sequential model)

Đôi lúc còn được gọi là mô hình kinh điển (classic model) hay mô hình thác nước (waterfall model). Mô hình này xem quá trình xây dựng một sản phẩm phần mềm bao gồm nhiều giai đoạn tách biệt, sau khi hoàn tất một giai đoạn thì chuyển đến giai đoạn sau.

Tuy vậy, thường thì các dự án có hàng ngàn trang tài liệu mà không ai ngoại trừ tác giả đọc đến nó. Thông tin ứng dụng chỉ nằm trong đầu mọi người và việc trao đổi thông tin là một trở ngại lớn để có được thành công của hệ thống. Kết luận là văn bản không phải là một phương tiện tốt để mô tả các yêu cầu phức tạp của ứng dụng. Thêm vào đó, mô hình bộc lộ một số nhược điểm quan trọng như:

- Mỗi qua hệ giữa các giai đoạn không được thể hiện
- Hệ thống phải được kết thúc ở từng giai đoạn do vậy rất khó thực hiện được đầy đủ những yêu cầu của khách hàng ..

Mô hình này được tóm tắt như sau:

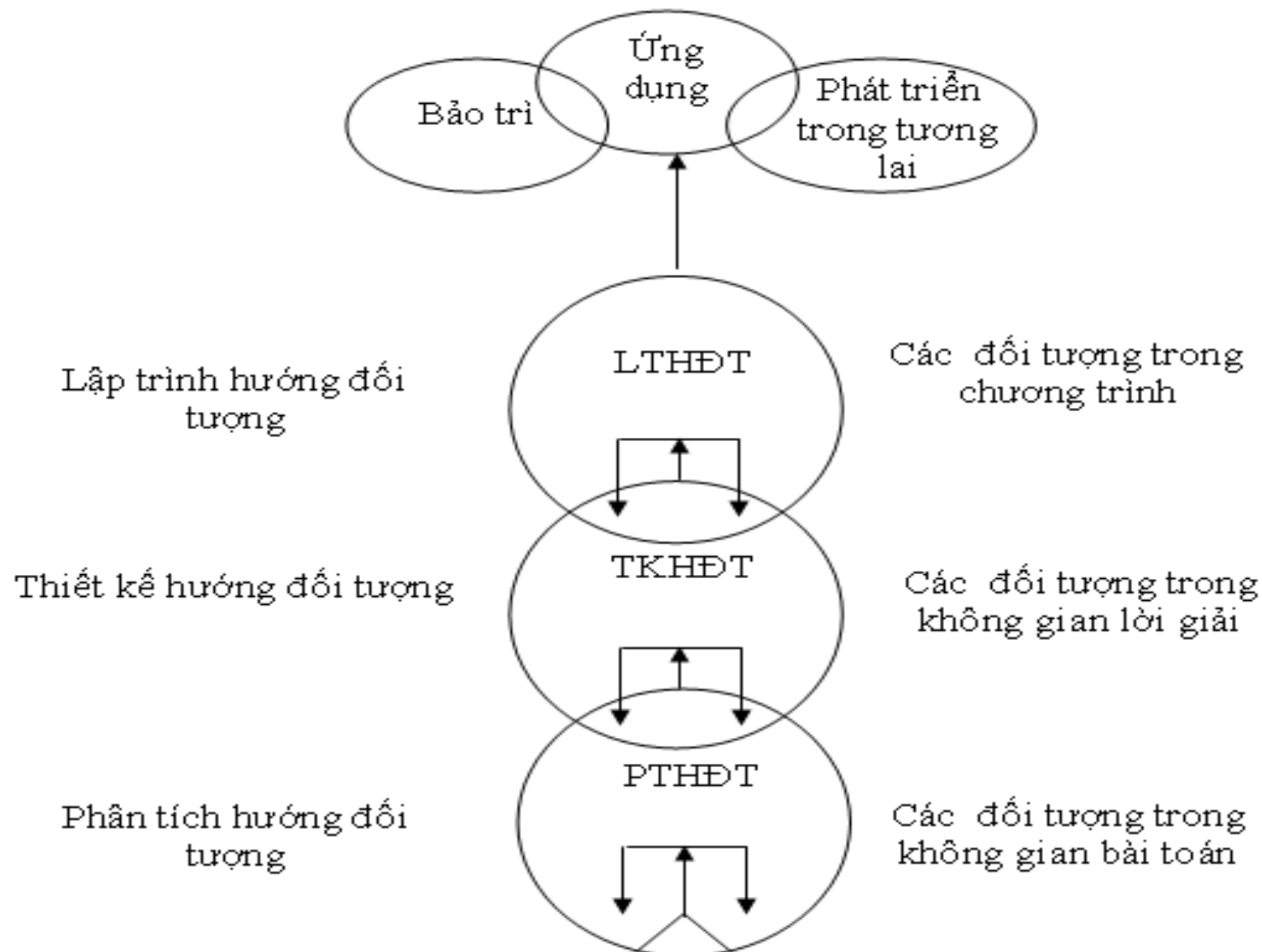


## 1.5.2. Mô hình đại phun nước

Đây là mô hình của cách tiếp cận hướng đối tượng, hệ thống được xem như là một hệ thống các thực thể tác động qua lại để đạt được một mục đích nào đó. Mô hình này tương ứng với mô hình thác nước trong cách tiếp cận hướng thủ tục ở trên. Ở đây, ta thấy trong có những phần lặp và giao nhau giữa các bước phân tích, thiết kế và cài đặt.



Các điểm chính của mô hình được tóm tắt như sau:

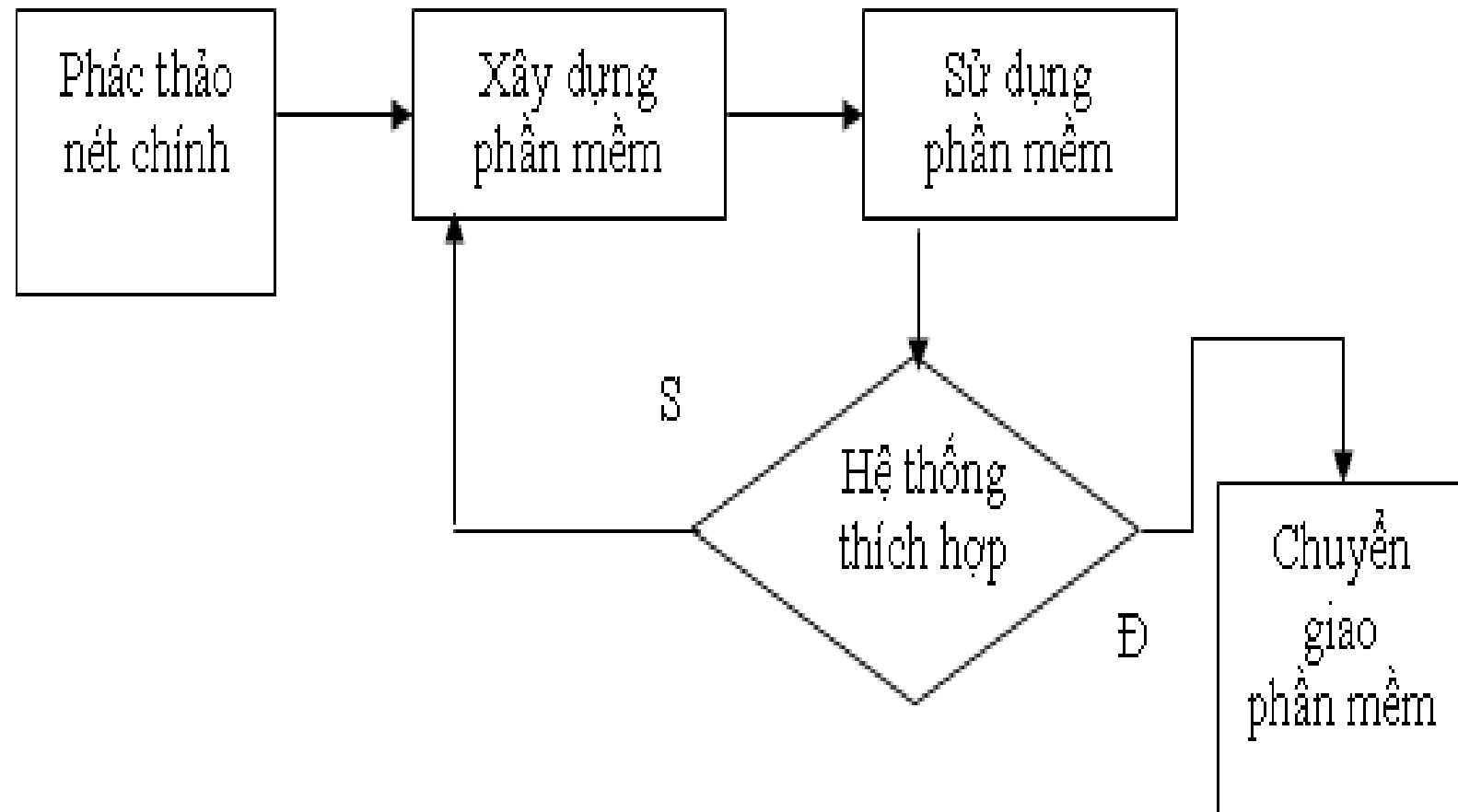


### 1.5.3 Mô hình mẫu (Prototyping model)

Thông thường, khách hàng sẽ đưa ra mục tiêu của họ một cách chung chung mà họ không biết hoặc không đưa ra một cách cụ thể những cái vào, cái ra và các tiến trình xử lý chúng. Thêm vào đó, chúng ta cũng không thể không quan tâm đến thuật toán sử dụng, tính tương thích của sản phẩm phần mềm với môi trường của nó như: phần cứng, hệ điều hành... Trong trường hợp này, mô hình mẫu có thể là sự lựa chọn tốt hơn cho người lập trình.

Mô hình mẫu là một cách để phá vỡ sự khắt khe, cứng nhắc trong chu trình tuần tự của dự án. Tuy vậy, trong mô hình mẫu, sử dụng sai làm hỏng phân tích và thiết kế, không bao giờ hoàn thiện được mẫu thành các ứng dụng thực sự là các vấn đề cần quan tâm. Thêm vào đó là hệ thống có thể không bao giờ được chuẩn hóa, chi tiết của việc xử lý, việc kiểm tra tính hợp lệ của dữ liệu và các đòi hỏi kiểm toán có thể bị bỏ quên trong việc đưa mẫu vào sản xuất.

Những điểm chính của mô hình mẫu được tóm tắt theo sơ đồ sau:



#### 1.5.4. Mô hình xoắn ốc (The spiral model)

Mô hình này được Boehm đưa ra nên đôi lúc còn được gọi là mô hình Boehm's (The Boehm's spiral model). Nó có thể xem là sự kết hợp giữa mô hình thác nước và mô hình mẫu và đồng thời thêm một thành phần mới - phân tích rủi ro. Bao gồm bốn hoạt động chính:

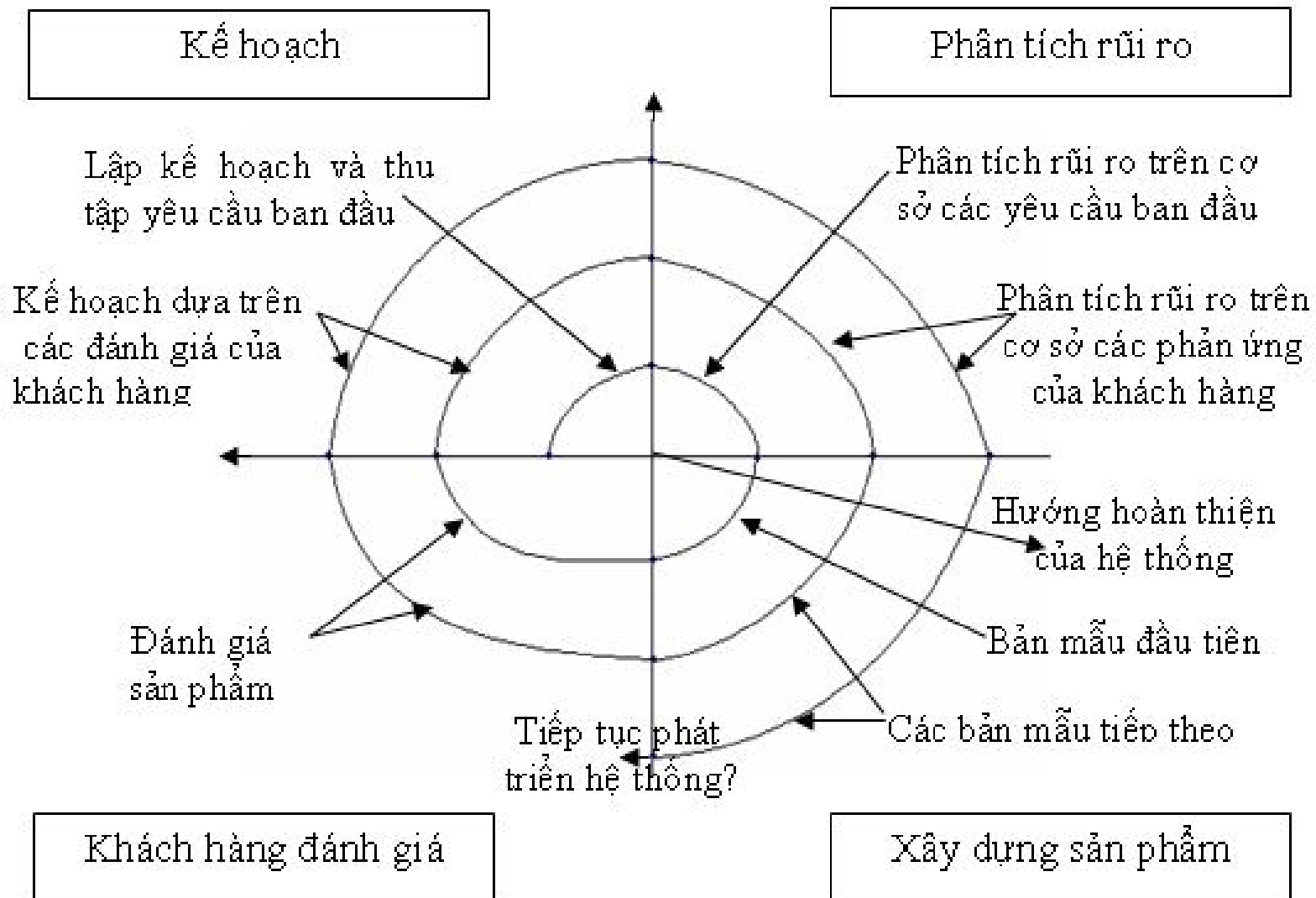
- Planning: Xác định mục tiêu, tương tác và ràng buộc.
- Risk analysis: Phân tích các lựa chọn và các chỉ định/giải quyết rủi ro.
- Engineering : Phát triển sản phẩm
- Customer evaluation: Đánh giá kết quả xây dựng.

Trong vòng đầu tiên của xoáy ốc, mục đích, lựa chọn, các ràng buộc được định nghĩa và các nguy cơ được xác định và phân tích. Nếu phân tích các lỗi chỉ ra rằng có một vài yêu cầu không chắc chắn, tạo mẫu có thể được tiến hành để giúp đỡ nhà phát triển và khách hàng. Mô phỏng và các mô hình khác có thể được sử dụng để xác định vấn đề và làm mịn các yêu cầu.

Khách hàng đánh giá công việc và đưa ra các gợi ý. Trên cơ sở ý kiến đó, phần tiếp theo của lập kế hoạch và phân tích lỗi xuất hiện.

Mô hình xoáy ốc hiện nay là mô hình hướng tiếp cận hiện thực nhất để phát triển các hệ thống lớn. Nó sử dụng mô hình mẫu như là cơ chế loại trừ lỗi, cho phép nhà phát triển áp dụng mô hình mẫu tại mỗi chu trình phát triển. Nó kế thừa cách tiếp cận hệ thống từng bước từ chu kỳ sống cổ điển nhưng kết hợp với quá trình lặp lại phù hợp với thực tế.

Mô hình được tóm tắt như sau:



### 1.5.5. Mô hình phát triển dựa trên thành phần

Xuất phát từ quan điểm: "Buy do not build", tư tưởng của phát triển dựa trên thành phần là lắp ráp hệ thống từ những thành phần đã có. Do vậy, kiến trúc phần mềm của hệ thống dựa vào kiến trúc phần mềm của các thành phần phần mềm tiêu chuẩn nên hệ thống đạt chất lượng cao hơn.

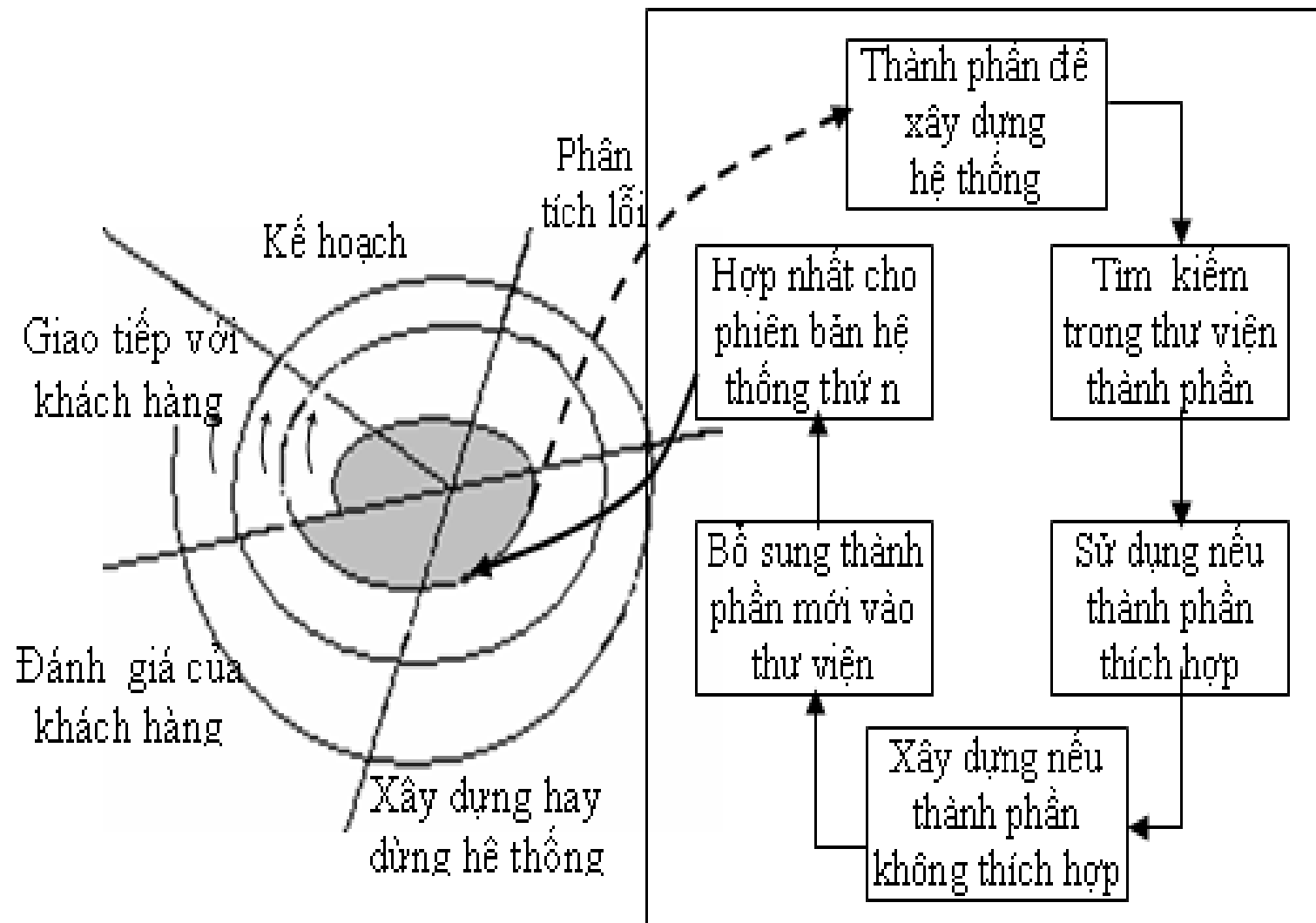
Phương pháp phát triển dựa trên thành phần gần tương tự như phương pháp phát triển hướng đối tượng. Hoạt động công nghệ bắt đầu với sự chỉ ra các lớp tham dự để phát triển hệ thống. Nếu các lớp này được tìm thấy trong thư viện và sự thích nghi là tốt, chúng sẽ được lấy ra và phát triển hệ thống. Ngược lại, chúng sẽ được phát triển để sử dụng và bổ sung vào thư viện sử dụng lại.

Thành phần phần mềm được sử dụng lại có độ chính xác cao và có thể nói là không chứa lỗi. Mặc dầu không thường xuyên được chứng minh về mặt hình thức nhưng với việc sử dụng lại, lỗi được tìm thấy và loại trừ, chất lượng của thành phần được cải thiện như là một kết quả.

Khi những thành phần sử dụng lại được ứng dụng thông qua tiến trình phần mềm, chúng ta ít tốn thời gian để tạo ra kế hoạch, mô hình, tài liệu, mã và dữ liệu mà chúng là cần thiết để tạo ra hệ thống. Thêm vào, chức năng cùng mức được phân phối cho người sử dụng với đầu vào ít công sức hơn, do vậy, hiệu suất phần mềm được cải thiện.



Những điểm chính của mô hình được tóm tắt như sau:



## 1.4. VAI TRÒ CỦA NGƯỜI DÙNG TRONG GIAI ĐOẠN PHÁT TRIỂN PHẦN MỀM

Trong những ứng dụng trước kia được xây dựng thường xuyên không có sự bàn bạc với người sử dụng, sự cô lập của các công nghệ phần mềm đối với người dùng dẫn đến những hệ thống có khả năng làm việc về mặt kỹ thuật, nhưng thông thường không đáp ứng được nhu cầu của người sử dụng, và thường xuyên làm gián đoạn quá trình làm việc.

Có rất nhiều lợi ích từ việc tham gia của người sử dụng trong phát triển ứng dụng.

- Trước tiên nó xây dựng sự cam kết của những người sử dụng - những người đương nhiên đảm nhiệm quyền sở hữu của hệ thống.
- Thứ hai, những người sử dụng là những chuyên gia thực sự của những công việc đang được tự động - lại được đại diện hoàn toàn thông qua sự phát triển.
- Thứ ba, những nhiệm vụ được người sử dụng thực hiện bao gồm việc thiết kế màn hình, các mẫu, các báo cáo, sự phát triển tài liệu của người sử dụng, sự phát triển và tiến hành của các cuộc kiểm tra công nhận,...