



**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

*Nhóm chuyên môn Nhập môn Công nghệ phần mềm*

# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

**Một số mô hình phát triển phần mềm phổ biến**



**1. Phân tích mô hình phát triển  
phần mềm phổ biến**

**2. Tổng kết**

*Sau bài học này, người học có thể:*

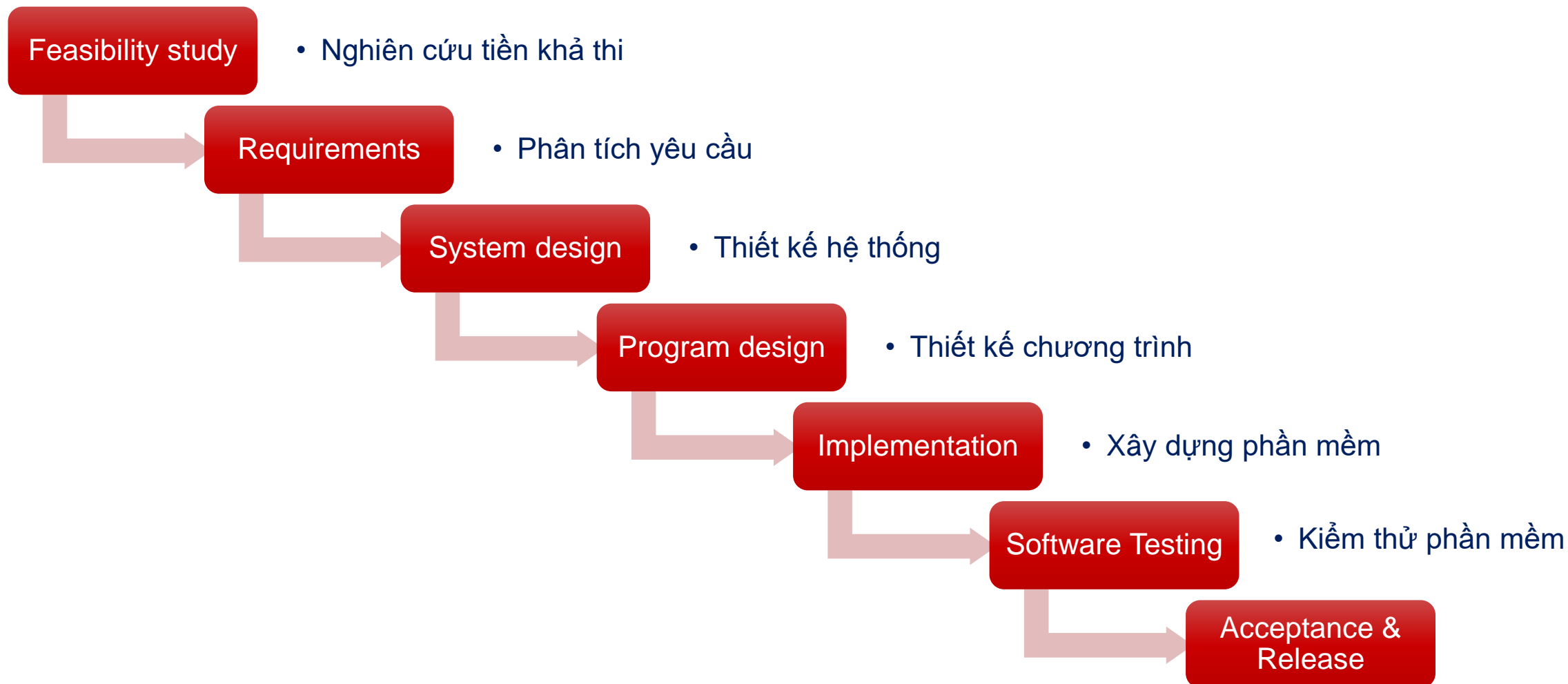
1. Biết **một số mô hình** phát triển phần mềm phổ biến
2. Hiểu về **sự khác biệt** của các mô hình phát triển phần mềm
3. Có cái nhìn tổng quan về các mô hình phát triển phần mềm nói chung

## 1. Phân tích mô hình phần mềm phổ biến

## 2. Tổng kết

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.1 Mô hình thác nước



*Các bước trong mô hình thác nước (waterfall)*

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



## 1.1 Mô hình thác nước

- Mô hình thác nước là mô hình vòng đời lâu đời nhất; được đề xuất bởi Winston Royce vào năm 1970.
- Mô hình này được gọi là thác nước vì nó thường được vẽ với một chuỗi các hoạt động qua các giai đoạn của vòng đời “xuống dốc” từ trái sang phải: phân tích, yêu cầu, đặc tả, thiết kế, cài đặt, kiểm thử, bảo trì
- Có nhiều phiên bản của mô hình thác nước:
  - Các giai đoạn / hoạt động có thể được cấu trúc theo các mức độ chi tiết khác nhau
  - Phản hồi có thể linh hoạt hơn hoặc ít hơn

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



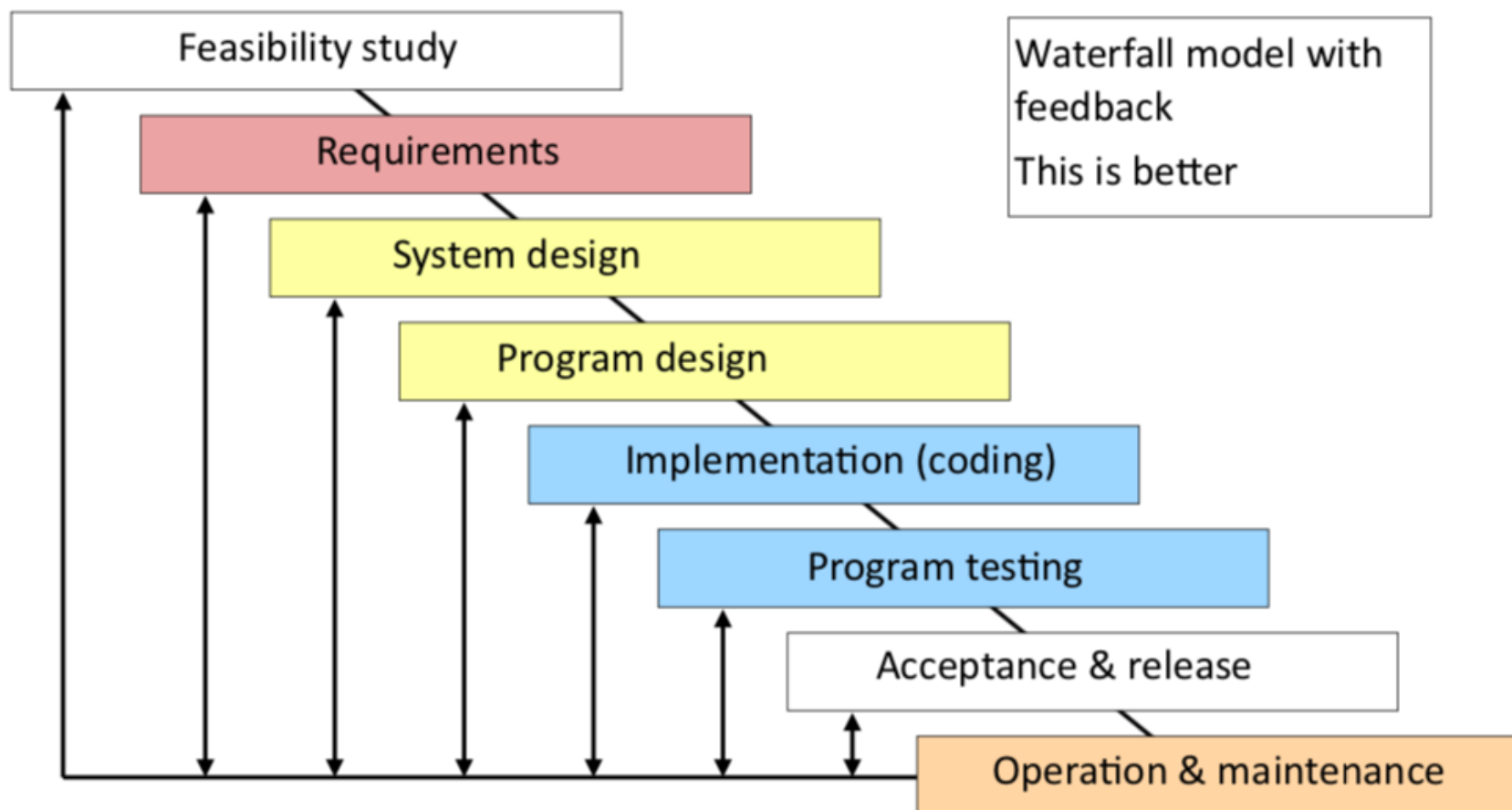
## 1.1 Mô hình thác nước

- Ưu điểm:
  - Tách nhiệm vụ trong mỗi giai đoạn 1 cách riêng biệt
  - Khả năng hiển thị tốt, dễ theo dõi
  - Quy trình kiểm soát chất lượng ở mỗi bước
  - Giám sát chi phí ở từng bước
- Nhược điểm:
  - Phụ thuộc vào các yêu cầu được xác định sớm từ đầu
  - Không khả thi trong một số trường hợp đòi hỏi có nhiều thay đổi
  - Trong thực tế, mỗi giai đoạn trong quy trình đều những cải tiến, phản hồi mới về các giai đoạn trước đó, thường đòi hỏi phải sửa đổi, điều chỉnh các giai đoạn trước đó.

**Mô hình thác nước không đủ linh hoạt**

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.1 Mô hình thác nước sửa đổi



Hình 1.1: Các bước trong mô hình thác nước cải tiến (Modified waterfall)



# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



## 1.1 Mô hình thác nước sửa đổi

Mô hình này thường được áp dụng cho:

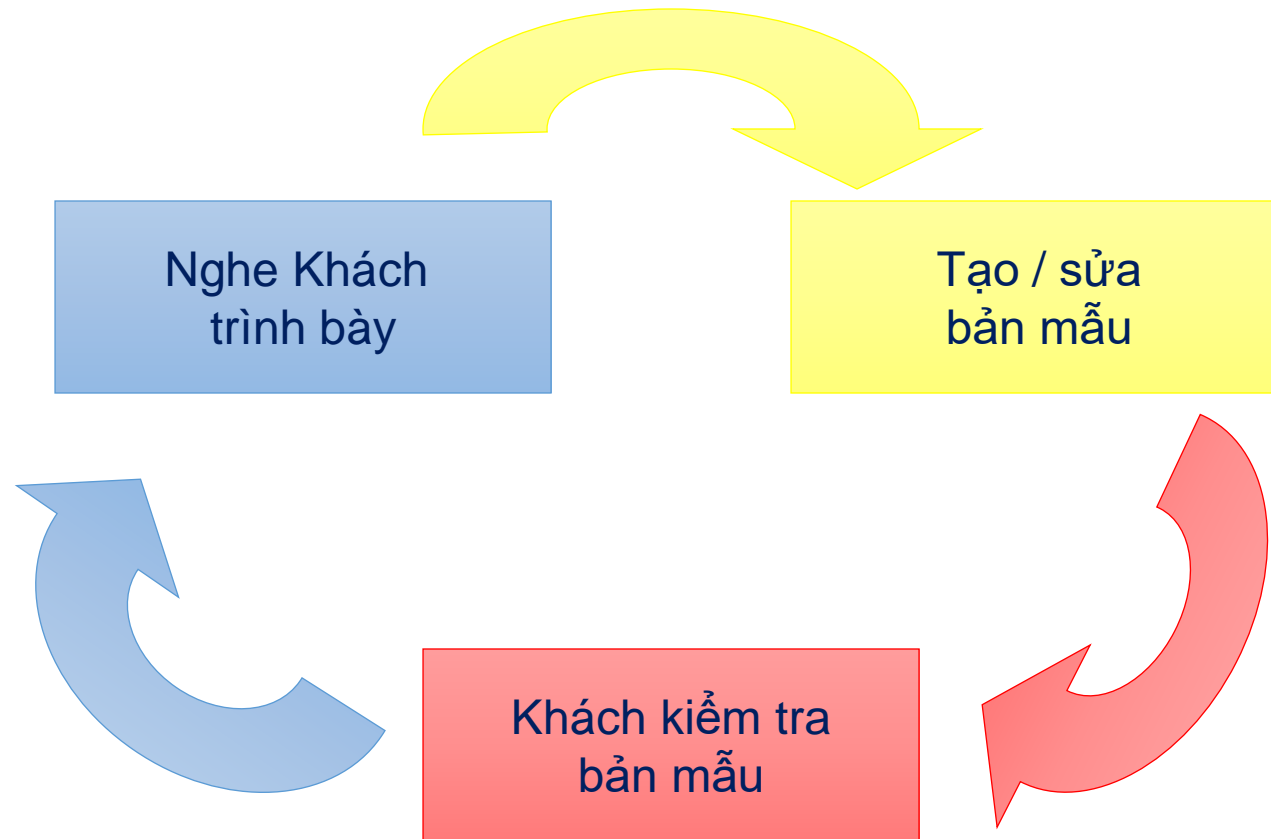
1. Mô hình thác nước đã sửa đổi hoạt động tốt nhất khi các yêu cầu được hiểu rõ và thiết kế đơn giản

*Ví dụ:*

- Chuyển đổi hệ thống xử lý dữ liệu thủ công trong đó các yêu cầu đã được hiểu rõ
  - Phiên bản mới của hệ thống có chức năng gần giống với sản phẩm trước đó
2. Các phần của một hệ thống lớn trong đó một số thành phần có yêu cầu được xác định rõ ràng và được tách biệt rõ ràng với phần còn lại của hệ thống

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.2 Mô hình mẫu thử (prototyping model)



Hình 1.2: Mô hình mẫu thử

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



## 1.2 Mô hình mẫu thử – Khi nào?

- Khi mới rõ mục đích chung chung của phần mềm, chưa rõ chi tiết đầu vào hay xử lý ra sao hoặc chưa rõ yêu cầu đầu ra
- Dùng để thu thập yêu cầu qua các thiết kế nhanh
- Các giải thuật, giải pháp kỹ thuật dùng làm bản mẫu có thể chưa nhanh, chưa tốt, miễn là có mẫu để thảo luận gợi ý yêu cầu của người dùng

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



## 1.3 Mô hình tăng trưởng

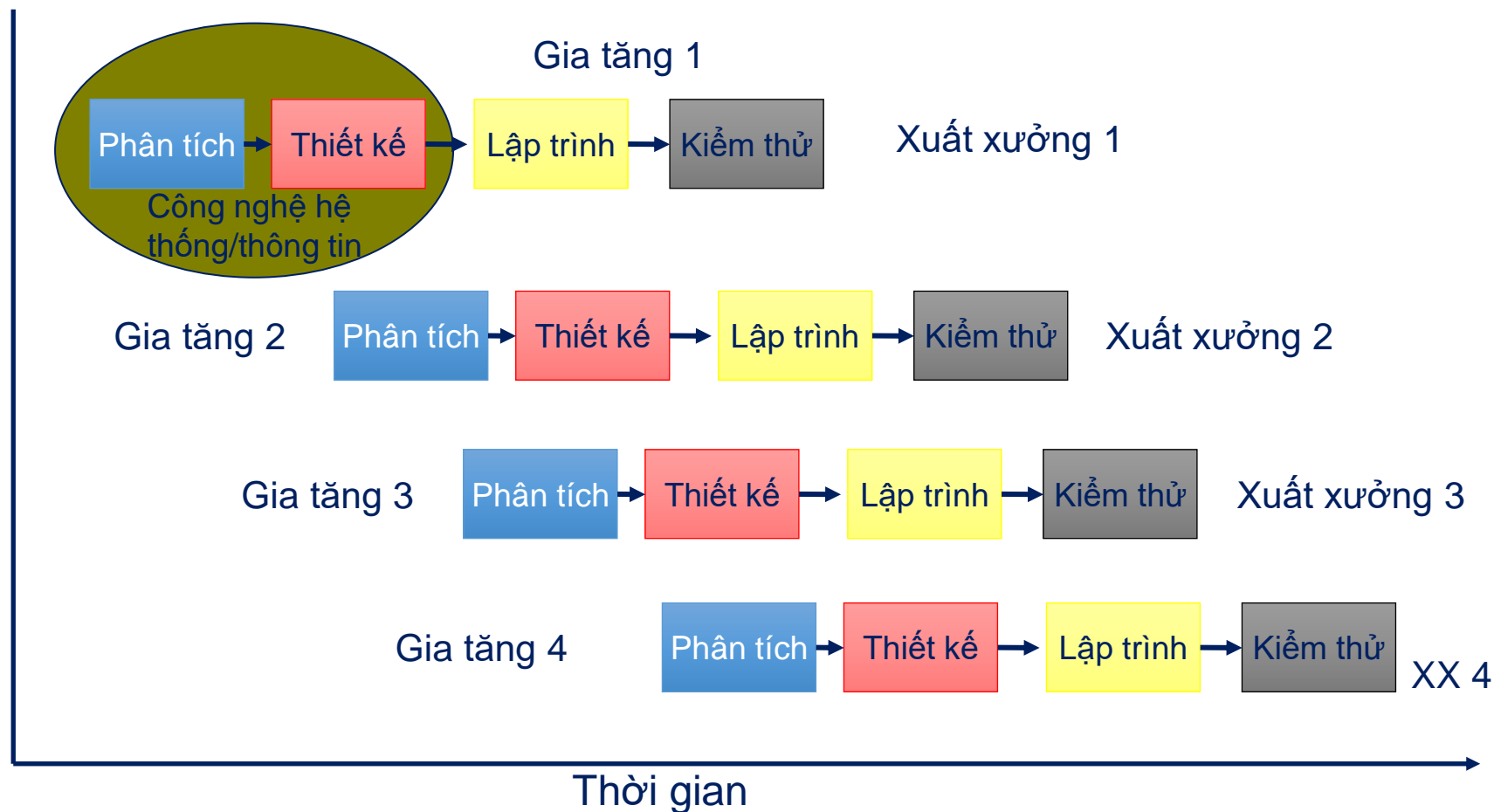
- Phần lớn các hệ phần mềm phức tạp đều tiến hóa theo thời gian: môi trường thay đổi, yêu cầu phát sinh thêm, hoàn thiện thêm chức năng, tính năng
- Các mô hình tiến hóa có tính lặp lại. Kỹ sư phần mềm tạo ra các phiên bản (versions) ngày càng hoàn thiện hơn, phức tạp hơn
- Các mô hình tiêu biểu:
  - Gia tăng (Incremental)
  - Xoắn ốc (Spiral)
  - Xoắn ốc WINWIN (WINWIN spiral)
  - Phát triển đồng thời (Concurrent development)

## 1.4 Mô hình gia tăng (incremental model)

- Kết hợp mô hình tuần tự và ý tưởng lặp lại của chế bản mẫu
- Sản phẩm với những yêu cầu cơ bản nhất của hệ thống được phát triển
- Các chức năng với những yêu cầu khác được phát triển thêm sau (gia tăng)
- Lặp lại quy trình để hoàn thiện dần

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.4 Mô hình gia tăng (incremental model)



Hình 1.3: Mô hình gia tăng

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---

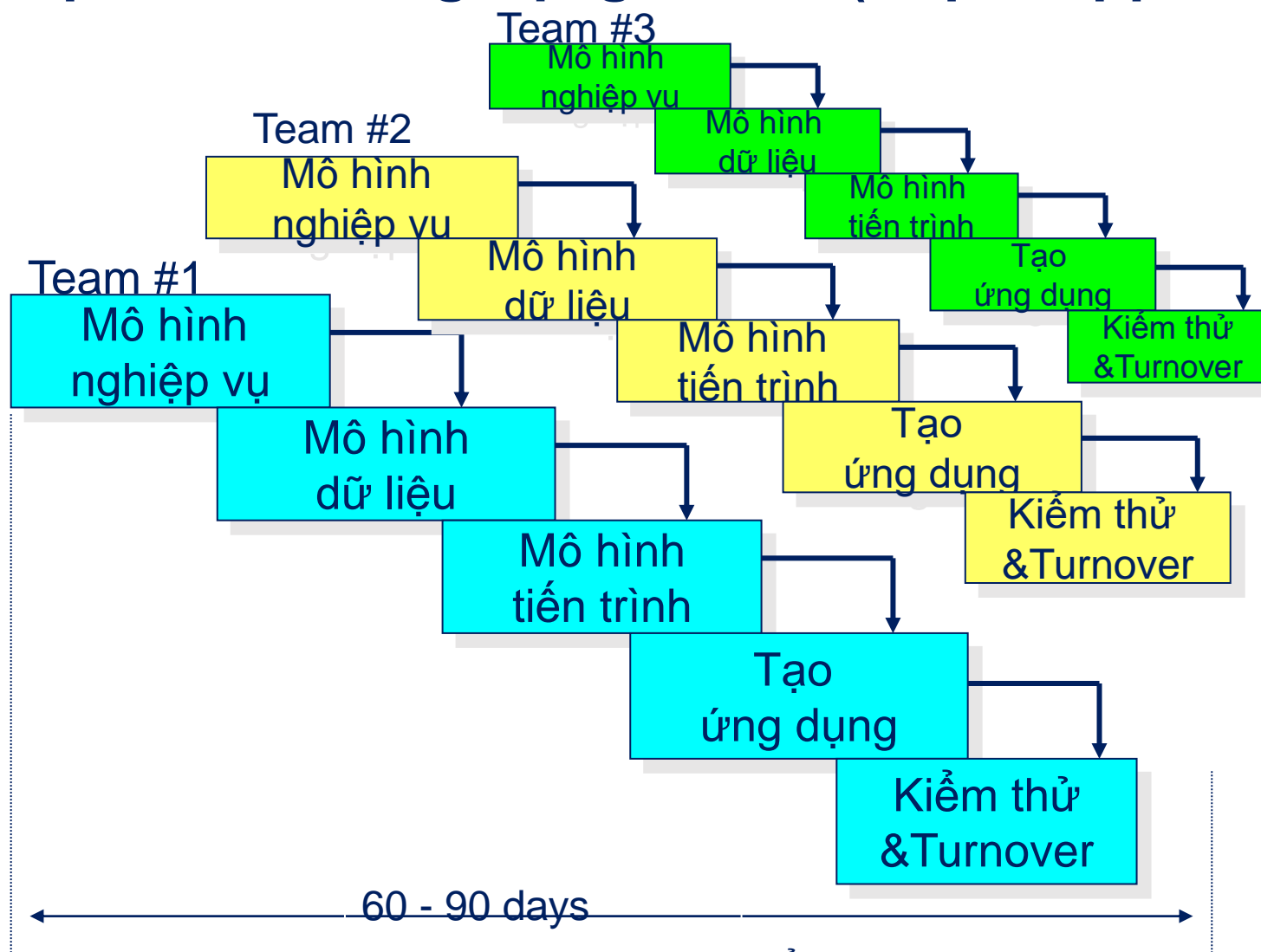


## 1.5 Mô hình phát triển ứng dụng nhanh (Rapid Application Development)

- Là quy trình phát triển phần mềm gia tăng, tăng dần từng bước (Incremental software development) với mỗi chu trình phát triển rất ngắn (60-90 ngày)
- Xây dựng dựa trên hướng thành phần (Component-based construction) với khả năng tái sử dụng (reuse)
- Gồm một số nhóm (teams), mỗi nhóm làm 1 RAD theo các pha: Mô hình nghiệp vụ, Mô hình dữ liệu, Mô hình xử lý, Tạo ứng dụng, Kiểm thử và đánh giá (Business, Data, Process, Appl. Generation, Test)

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.5 Mô hình phát triển ứng dụng nhanh (Rapid Application Development)

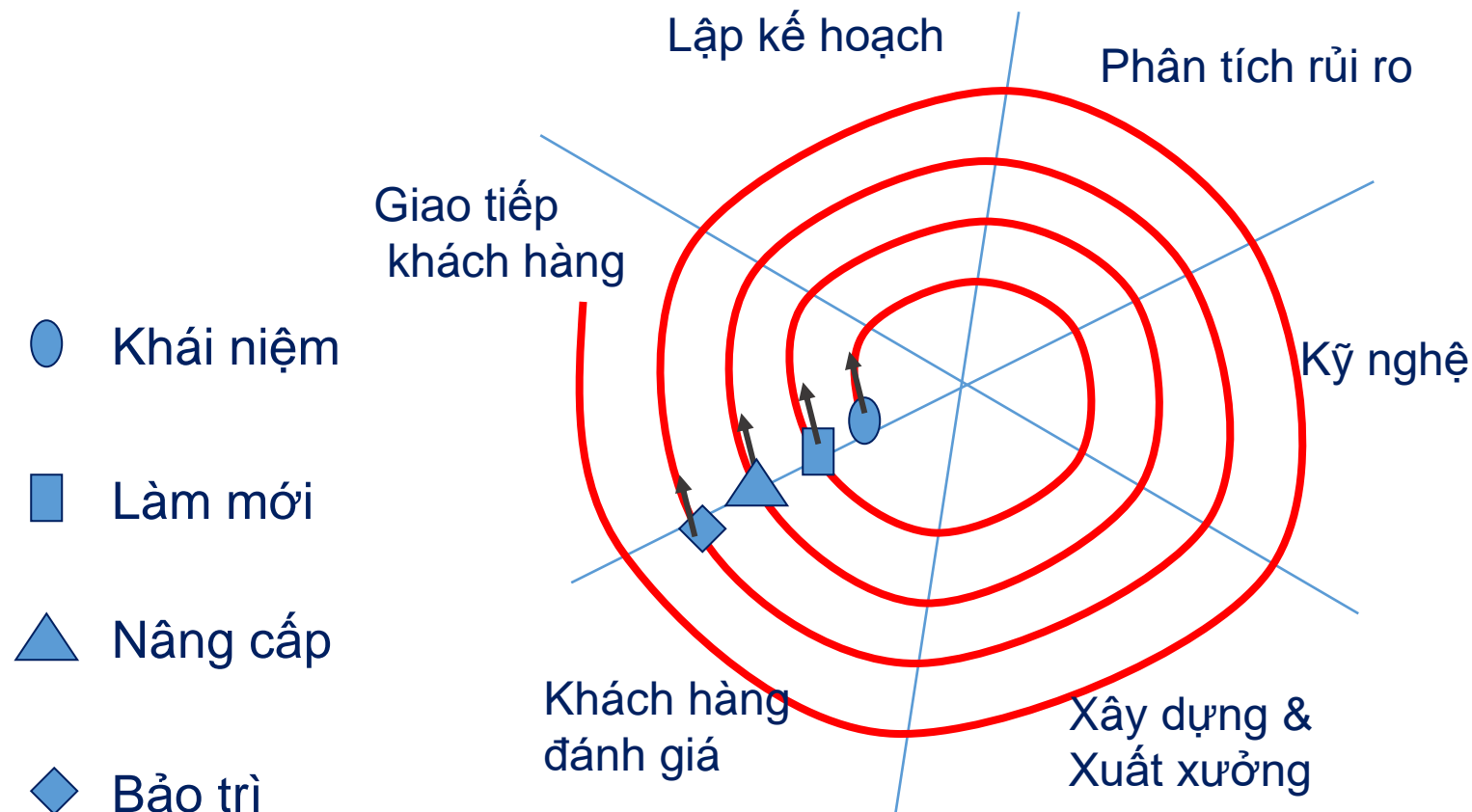


Hình 1.4: Mô hình phát triển ứng dụng nhanh



# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

## 1.6 Mô hình xoắn ốc (spiral)



Hình 1.5: Mô hình phát xoắn ốc

# 1. PHÂN TÍCH MÔ HÌNH PHẦN MỀM PHỔ BIẾN

---



## 1.6 Mô hình xoắn ốc

- Giao tiếp khách hàng: giữa người phát triển và khách hàng để tìm hiểu yêu cầu, ý kiến
- Lập kế hoạch: Xác lập tài nguyên, thời hạn và những thông tin khác
- Phân tích rủi ro: Xem xét mạo hiểm kỹ thuật và mạo hiểm quản lý
- Kỹ nghệ: Xây dựng một hay một số biểu diễn của ứng dụng
- Xây dựng và xuất xưởng: xây dựng, kiểm thử, cài đặt và cung cấp hỗ trợ người dùng (tư liệu, huấn luyện, . . .)
- Đánh giá của khách hàng: Nhận các phản hồi của người sử dụng về biểu diễn phần mềm trong giai đoạn kỹ nghệ và cài đặt

## 1.6 Mô hình xoắn ốc

- **Mô hình này thường được áp dụng:**
  - Tốt cho các hệ phần mềm quy mô lớn
  - Dễ kiểm soát các mạo hiểm ở từng mức tiến hóa
  - Khó thuyết phục khách hàng là phương pháp tiến hóa xoắn ốc có thể kiểm soát được
  - Chưa được dùng rộng rãi như các mô hình tuyến tính hoặc chế thử

1. Phân tích mô hình phần mềm phổ biến

## 2. Tổng kết

## 2. TỔNG KẾT

---

- Không có một quy trình phát triển phần mềm nào là hoàn hảo cho tất cả các dự án.
- Mỗi quy trình có ưu điểm và hạn chế riêng, và nó cũng có thể cần được điều chỉnh hoặc kết hợp để phù hợp với yêu cầu cụ thể của dự án và nhóm phát triển.
- Việc lựa chọn quy trình phù hợp với dự án là một bước quan trọng để đảm bảo sự thành công của dự án phát triển phần mềm.

1. Bài học đã cung cấp cho người học một số **mô hình phát triển phần mềm phổ biến** cũng như các đặc điểm của nó. Thông qua các mô hình này, người học có thể hình dung tổng quan về các mô hình phát triển phần mềm
2. Tiếp sau bài này, người học có thể tự tìm hiểu kĩ hơn về các mô hình phát triển phần mềm khác

# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

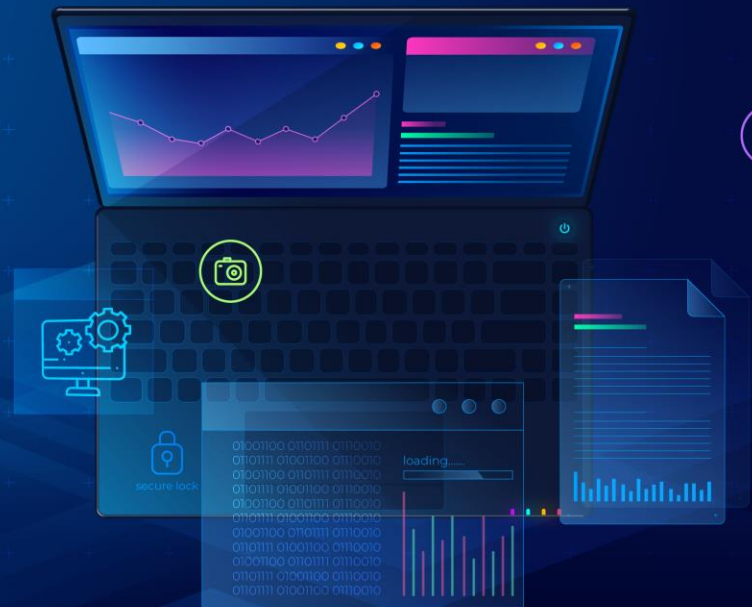
## Một số quy trình phát triển phần mềm phổ biến

Biên soạn:

TS. Nguyễn Nhất Hải

Trình bày:

TS. Nguyễn Nhất Hải



# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

*Bài học tiếp theo:*

## Hướng dẫn bài tập: Chương 2. Vòng đời phần mềm

Tài liệu tham khảo:

- [1] R. Pressman, Software Engineering: A Practitioner's Approach. 8th Ed., McGraw-Hill, 2016 và bộ slide đi kèm.
- [2] I. Sommerville, Software Engineering. 10th Ed., AddisonWesley, 2017.
- [3] Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Ed., Springer.
- [4] Shari Lawrence Pleege, Joanne M. Atlee, Software Engineering theory and practice. 4th Ed., Pearson, 2009