

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

PHẦN I – TỔNG QUAN VỀ CÔNG NGHỆ PHẦN MỀM

Bộ môn Công nghệ phần mềm,
Khoa CNTT&TT, Đại học Cần Thơ



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

1

Nội dung

- Giới thiệu về Công nghệ phần mềm
- Các mô hình về tiến trình phần mềm
- Quản lý phần mềm



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

2

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

I.2 – CÁC MÔ HÌNH VỀ TIẾN TRÌNH PHẦN MỀM



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

3

Câu hỏi

- Làm cách nào để tạo ra được một phần mềm?
- Các thức để tạo ra một phần mềm được áp dụng giống như nhau cho mọi sản phẩm phần mềm?



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

4

CÁC MÔ HÌNH VỀ TIẾN TRÌNH PHẦN MỀM



- Tiến trình
- Một số mô hình về tiến trình phần mềm

Tiến trình (Process)



- **Định nghĩa**
 - **Tiến trình:** một chuỗi các bước bao gồm các hoạt động, các ràng buộc và các tài nguyên mà chúng tạo ra kết quả được mong đợi.
- **Tiến trình** bao gồm một bộ các công cụ và các kỹ thuật.

Tiến trình



- **Các đặc trưng của tiến trình**

- Quy định tất cả các hoạt động của tiến trình chính.
- Sử dụng các nguồn tài nguyên, phụ thuộc vào tập các ràng buộc (chẳng hạn như kế hoạch làm việc).
- Tạo ra các sản phẩm cuối cùng hoặc trung gian.
- Có thể được tạo thành từ các tiến trình con bằng hệ thống phân cấp hay các liên kết.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

7

Tiến trình



- **Các đặc trưng của tiến trình**

- Mỗi hoạt động của tiến trình có tiêu chuẩn vào và ra.
- Các hoạt động được tổ chức theo trình tự vì thế sự tính toán về thời gian là rõ ràng.
- Mỗi tiến trình có các nguyên tắc hướng dẫn, bao gồm các mục tiêu của từng hoạt động.
- Các ràng buộc có thể áp dụng vào một hoạt động, tài nguyên hay sản phẩm.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

8

Tiến trình



- **Tầm quan trọng của tiến trình**
 - Áp đặt cấu trúc và tính bền vững lên một tập các hoạt động.
 - Hướng dẫn ta hiểu, điều khiển, kiểm tra và cải thiện các hoạt động.
 - Cho phép ta có được các kinh nghiệm.

Tiến trình



- **Mô hình hóa tiến trình là gì?**
- **Lý do để mô hình hóa tiến trình?**

Tiến trình



- **Mô hình hóa tiến trình**

- Các mô tả cho cách thức tạo ra phần mềm theo cách được thực hiện trong thực tế.
- Các quy định cho cách thức phát triển phần mềm phải tiến bộ.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

11

Tiến trình



- **Lý do để mô hình hóa tiến trình**

- Hình thành một cách hiểu chung.
- Tìm ra sự không nhất quán, sự dư thừa hay thiếu.
- Tìm ra và đánh giá các hoạt động phù hợp để đạt được các mục tiêu của tiến trình.
- Cụ thể hóa một tiến trình chung cho một hoàn cảnh cụ thể.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

12

Tiến trình

Chu kỳ sống của phần mềm

- Khi một tiến trình liên quan tới việc xây dựng một phần mềm, tiến trình phát triển phần mềm có thể được xem như chu kỳ sống của phần mềm.



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

13

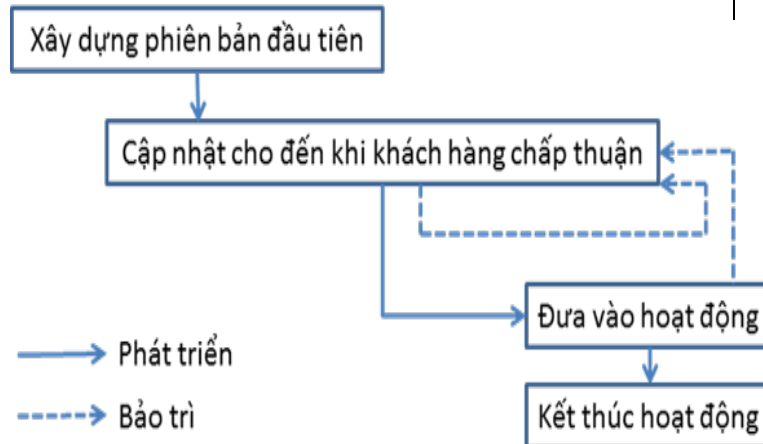
Một số mô hình về tiến trình phần mềm

- Mô hình xây dựng và hiệu chỉnh
- Mô hình thác nước
- Mô hình chữ V
- Mô hình bản mẫu
- Mô hình ứng dụng nhanh
- Mô hình gia tăng
- Mô hình xoắn ốc
- Mô hình RUP

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

14

Mô hình xây dựng và hiệu chỉnh (code-and-fix model)



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

15

Mô hình thác nước (Waterfall Model)

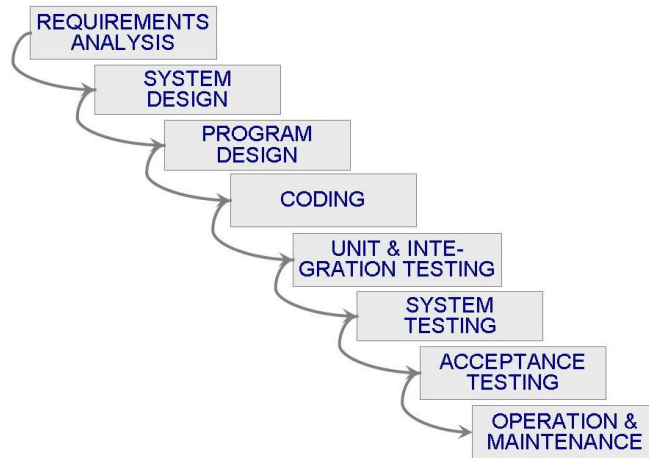


- Là một trong các mô hình đầu tiên về tiến trình phần mềm.
- Phù hợp với những bài toán được hiểu kỹ, có ít hay không có các thay đổi về yêu cầu.
- Đơn giản và dễ giải thích với khách hàng.
- Biểu diễn:
 - Một tổng quan mức rất cao của tiến trình phát triển.
 - Một chuỗi tuần tự các hoạt động của tiến trình.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

16

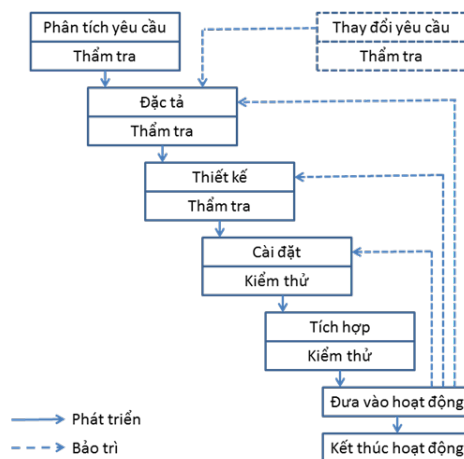
Mô hình thác nước



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

17

Mô hình thác nước



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

18

Mô hình thác nước



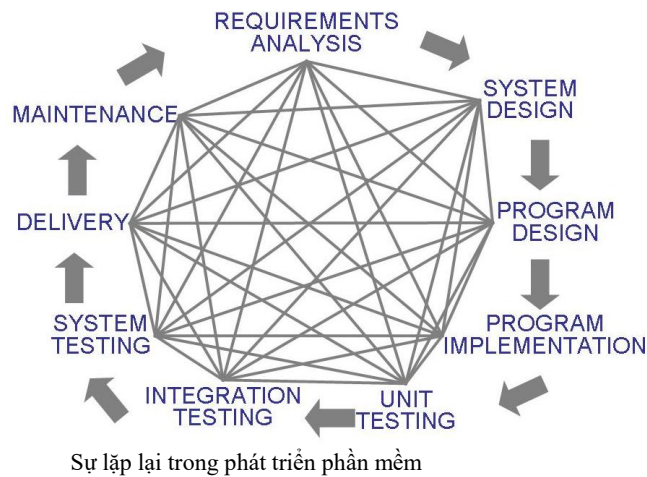
- Ưu điểm
 - Mô hình có ưu điểm là tính kỷ luật cao.
 - Quy định tốt về tài liệu cho mỗi giai đoạn.
 - Mỗi giai đoạn được kiểm chứng cẩn thận bởi nhóm SQA.

Mô hình thác nước



- Hạn chế
 - Có quá nhiều kiểm thử, thẩm tra và tài liệu hóa nên khó hình dung và khó hiểu đối với khách hàng.
 - Xem sự phát triển phần mềm như một tiến trình sản xuất hơn là tiến trình sáng tạo.
 - Không có các hoạt động lặp mà chúng đưa đến việc tạo ra sản phẩm cuối cùng.
 - Phải chờ đợi lâu trước khi có sản phẩm cuối.

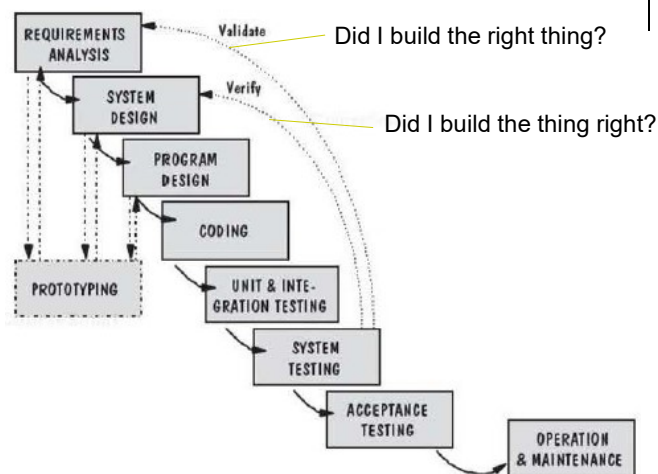
Mô hình thác nước Vs sự lặp lại trong phát triển phần mềm



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

21

Mô hình thác nước với bản mẫu (Waterfall model with prototyping)



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

22

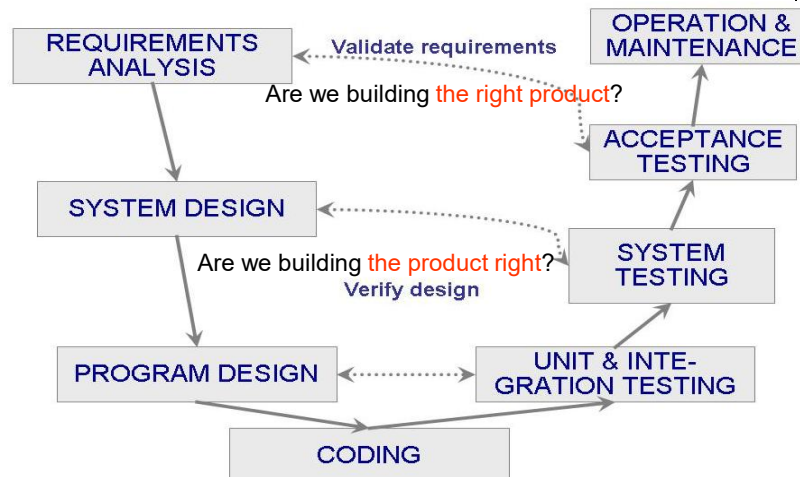
Mô hình chữ V (V Model)

- Là một sự biến đổi của mô hình thác nước.
- Sử dụng kiểm thử đơn vị để thẩm tra/kiểm tra (**verify**) thiết kế thủ tục.
- Sử dụng kiểm thử tích hợp để thẩm tra thiết kế hệ thống
- Sử dụng kiểm thử chấp nhận để công nhận hợp lệ/xác nhận (**validate**) các yêu cầu.
- Nếu các vấn đề được tìm thấy trong suốt sự thẩm tra và công nhận hợp lệ, phần bên trái của mô hình chữ V có thể được tái thực hiện trước khi việc kiểm thử phần bên phải được tái thực hiện.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

23

Mô hình chữ V



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

24

Mô hình chữ V



- Boehm phân biệt giữa validation và verification như sau:
 - Validation: Are we building **the right product**?
 - Verification: Are we building **the product right**?
- Theo Capability Maturity Model (CMMI-SW v1.1),
 - Validation: The process of evaluating software during or at the end of the development process to determine whether it **satisfies specified requirements**. [IEEE-STD-610]
 - Verification: The process of evaluating software to determine whether **the products of a given development phase satisfy the conditions imposed at the start of that phase**. [IEEE-STD-610]

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

25

Mô hình bản mẫu (Prototyping Model)

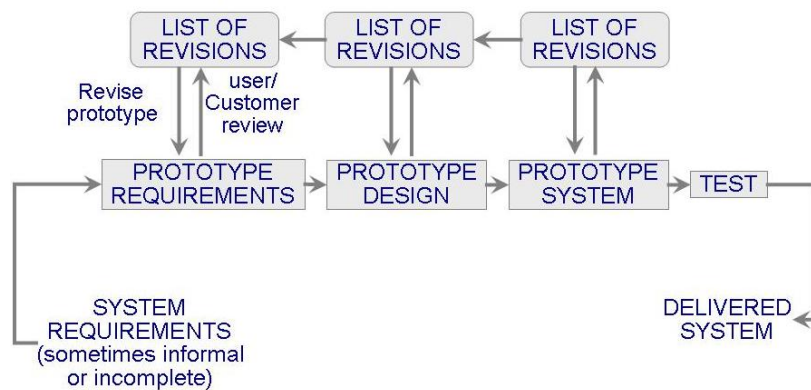


- Cho phép sự nghiên cứu về các yêu cầu và thiết kế được lặp lại.
- Giảm sự rủi ro và sự không chắc chắn trong phát triển.
- Sử dụng mô hình bản mẫu khi các yêu cầu không rõ ràng và cần minh họa cao về giao diện người dùng.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

26

Mô hình bản mẫu (Prototyping model)



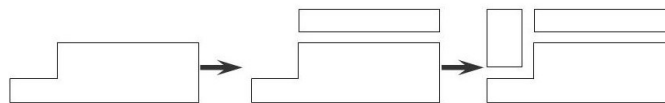
TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

27

Mô hình tăng trưởng (Incremental Model)



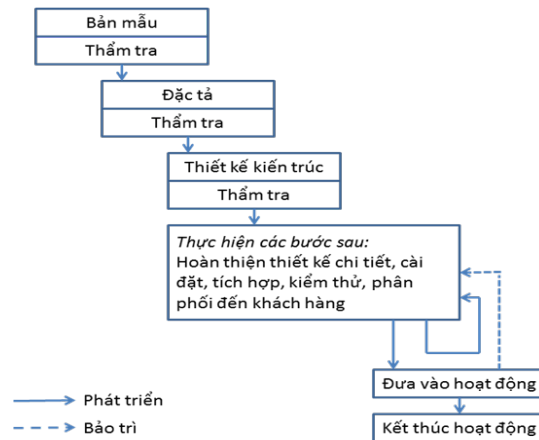
- Kết hợp mô hình tuần tự và ý tưởng lặp lại của chế bản mẫu.
- Sản phẩm lõi cho những yêu cầu cơ bản nhất của hệ thống được phát triển.
- Các chức năng cho những yêu cầu khác được phát triển thêm sau (tăng trưởng, gia tăng).
- Lặp lại quy trình để hoàn thiện dần.



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

28

Mô hình tăng trưởng

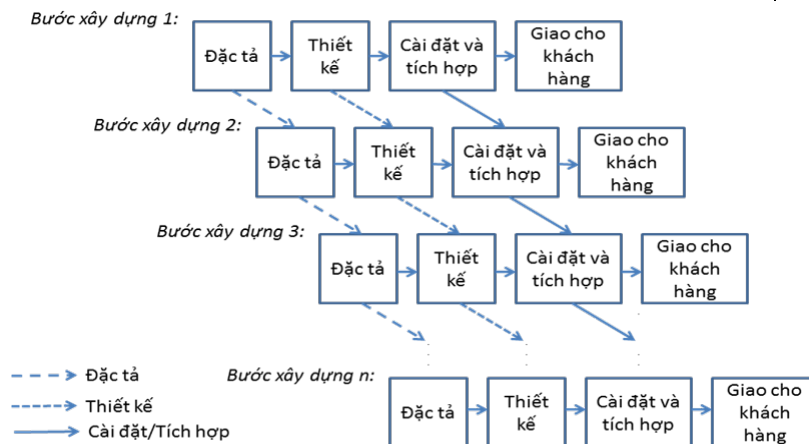


Mô hình tăng trưởng

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

29

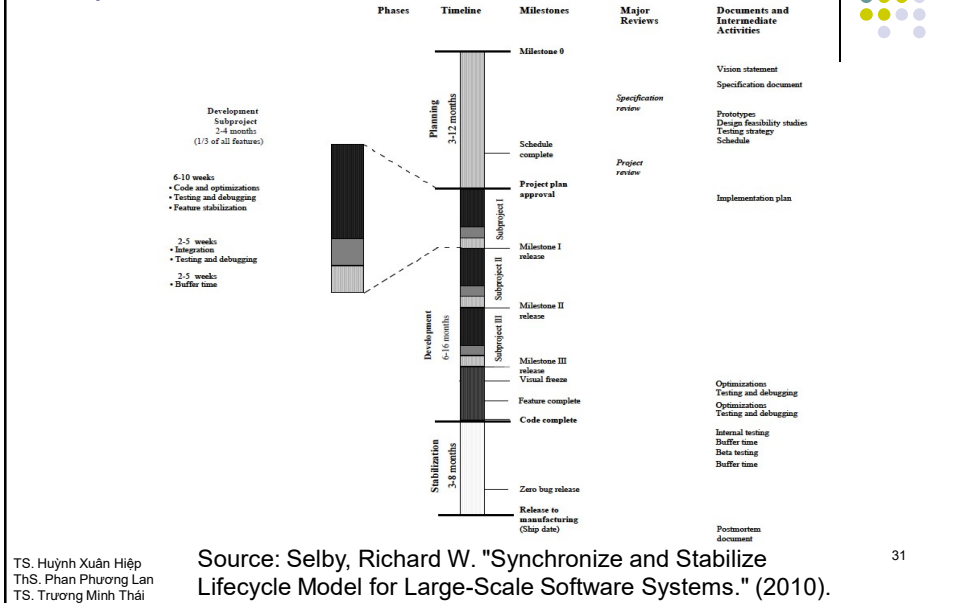
Mô hình tăng trưởng nhiều rủi ro (risk incremental model)



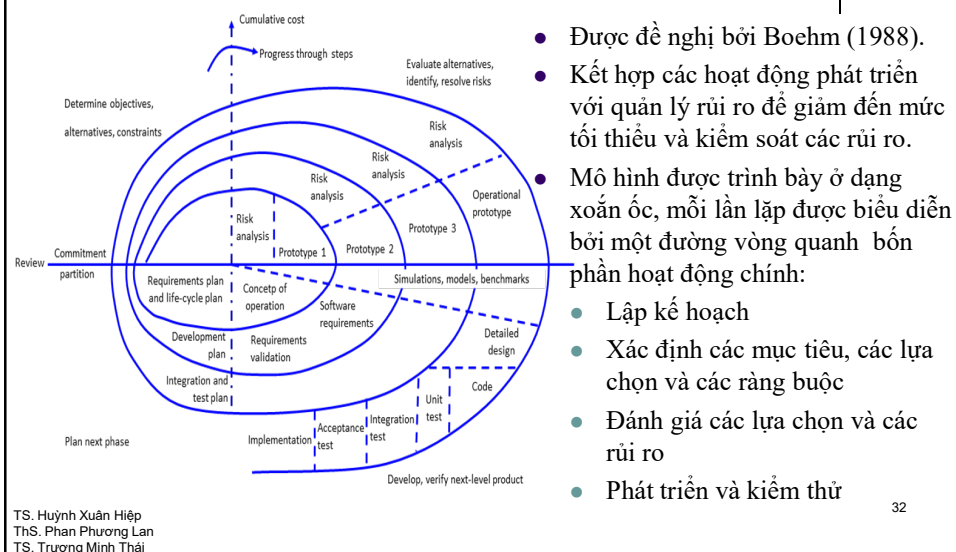
TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

30

Mô hình đồng bộ và ổn định (synchronize-and-stabilize model)



Mô hình xoắn ốc (Spiral Model)



Mô hình xoắn ốc



- Ưu điểm:
 - Các công việc luân phiên và chịu các ràng buộc đã hỗ trợ cho việc tái sử dụng phần mềm hiện có.
 - Đánh giá được mức độ rủi ro.
 - Mục tiêu quan trọng luôn là chất lượng phần mềm.
 - Giảm nhẹ kiểm thử và nhanh chóng sửa chữa những lỗi xảy ra.
- Nhược điểm:
 - Dành riêng cho các phần mềm nội bộ có kích thước lớn.
 - Kích thước sản phẩm ảnh hưởng đến giá thành việc phân tích rủi ro.

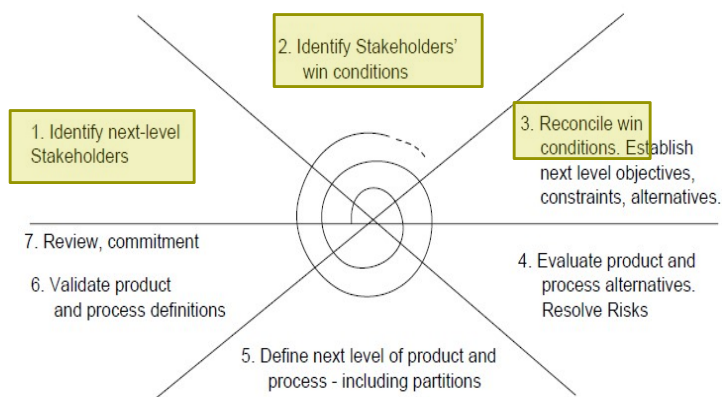
TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

33

Một dạng mở rộng của mô hình hình xoắn ốc Next Generation Process Model (NGPM)



- Theory W: Make Everyone a Winner
- 02 nguyên tắc cơ bản của lý thuyết W: “plan the flight and fly the plan” và “identify and manage your risks”

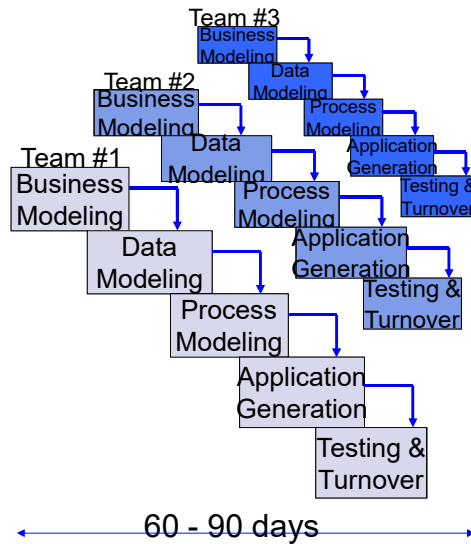


The NGPM Theory W extensions to the Spiral Model

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

34

Mô hình phát triển ứng dụng nhanh (Rapid Application Development: RAD)



- Là tiến trình phát triển phần mềm dạng tăng trưởng, tăng dần từng bước với mỗi chu kỳ phát triển rất ngắn (60-90 ngày).
- Xây dựng dựa trên hướng thành phần với khả năng tái sử dụng.
- Gồm một số nhóm, mỗi nhóm làm 1 RAD theo các pha: Mô hình hóa nghiệp vụ, Mô hình hóa dữ liệu, Mô hình hóa xử lý, Tạo ứng dụng, Kiểm thử và đánh giá (Business, Data, Process, Appl. Generation, Testing).

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

35

Mô hình phát triển ứng dụng nhanh - Thuận lợi

- Giảm thời gian phát triển.
- Tăng khả năng tái sử dụng của các thành phần
- Khuyến khích khách hàng phản hồi
- Tích hợp ngay từ ban đầu nên giải quyết rất nhiều vấn đề trong tích hợp.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

36

Mô hình phát triển ứng dụng nhanh - Bất lợi

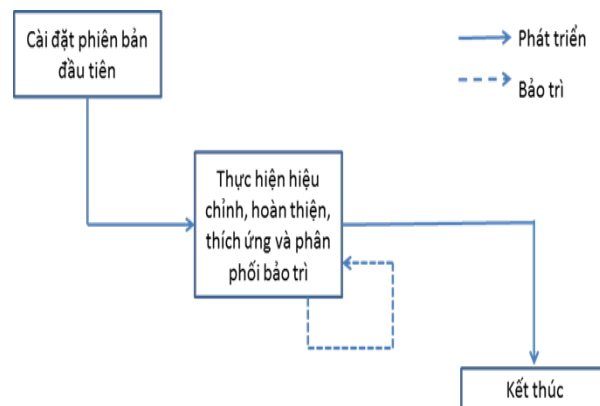


- Cần nguồn nhân lực dồi dào để tạo các nhóm cho các chức năng chính.
- Yêu cầu hai bên cam kết trong thời gian ngắn phải có phần mềm hoàn chỉnh, thiếu trách nhiệm của một bên dễ làm dự án đổ vỡ.
- RAD không phải tốt cho mọi ứng dụng, nhất là với ứng dụng không thể module hóa hoặc đòi hỏi tính năng cao.

TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

37

Mô hình Mã nguồn mở (*open source model*)



TS. Huỳnh Xuân Hiệp
ThS. Phan Phương Lan
TS. Trương Minh Thái

38

Đọc thêm



- *Mô hình lặp và tăng trưởng (iterative-and-incremental model)*
- *Mô hình cây tiến hóa (evolution-tree model)*
- *Mô hình phát triển nhanh (extreme programming and agile processes)*

HẾT PHẦN I.2

