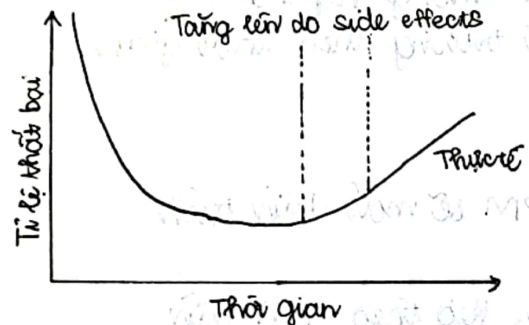


# CÁC ĐẶC TRƯNG CỦA PHẦN MỀM

④ Không mòn cũ nhưng thoái hoá theo thời gian



(Biểu đồ tỷ lệ thất bại theo thời gian của phần mềm)

Thuở ban đầu, khi mới phát triển phần mềm, sẽ có nhiều khuyết khuyết chưa được phát hiện ra, gây ra lỗi. Chúng sẽ được sửa chữa, bù đắp và cải tiến, từ đó ổn định hơn. Tuy nhiên, do môi trường sử dụng, nhu cầu thay đổi, phần mềm luôn không ngừng đổi theo và nâng cấp. Cũng thay đổi nhiều, lỗi sẽ càng phát sinh, phần mềm sẽ xuống cấp, thoái hoá (chưa giải quyết xong thì lại có thêm lỗi mới).

② KHÔNG ĐƯỢC LẮP RÁP TỬ MẪU CÓ SẴN

Khác với sản phẩm thông thường, chỉ có 1 thiết kế và sản xuất theo mẫu, sản phẩm phần mềm thường gặp những yêu cầu riêng (của khách hàng, doanh nghiệp), không ngừng phát triển để trở nên tốt hơn.

③ Luôn luôn thay đổi (thay đổi lõi bản chất)

Việc thay đổi, tiến hoá phần mềm là vô cùng quan trọng, nếu không sẽ không thể tồn tại, bởi những lý do sau:

- Môi trường tính toán thay đổi, công nghệ mới hình thành, phần mềm cần thích nghi
- Nghiệp vụ thay đổi, nhu cầu con người thay đổi.
- Lỗi phát sinh cần sửa chữa, hay có thêm tính năng mới, cần nâng cấp.
- Không cạnh tranh được sẽ không còn như hút nhai đầu H1.

④ Phức tạp, khó hiểu, vô hình

⑤ Được phát triển theo nhóm

Để phát triển phần mềm qua các giai đoạn, từng task được phân cho những bộ phận / cá nhân có năng lực đảm nhiệm mới có thể đạt được mục tiêu chung một cách hiệu quả, nhanh nhất và tối ưu, đồng thời giảm thiểu tối đa lỗi phát sinh.

# TIÊU CHÍ CỦA MỘT PHẦN MỀM TỐT

DANH MỤC

- Functionality**  
TM yêu cầu và HD đúng kỳ vọng
- Usability**  
Dễ hiểu và dễ dùng (với user)
- Chiếc**  
**mảnh**
- Performance**  
Nhanh, hiệu quả, thời gian phản hồi không quá lâu.
- Reliability**  
Đáng tin cậy, ít lỗi.
- Security**  
Bảo mật thông tin, chống tấn công.
- Portability**  
Chạy trên nền tảng khác nhau mà ít chỉnh sửa.
- Maintainability**  
Dễ dàng update, thay đổi, sửa lỗi, dễ nâng cấp trong tương lai.
- Scalability**  
Xử lý lượng lớn data, user, request, ...
- Testability**  
Dễ test (kiểm thử) có đúng yêu cầu hay không.

Đối tượng	
Chủ đầu tư	Kinh phí lời yêu tố quan trọng, TG hợp lý (≡ ngắn, ≡ dài)
Người dùng	Đáp ứng nhu cầu SD, dễ hiểu, dễ dùng, dễ tiếp cận, đáng tin cậy
Nhà phát triển	Dễ bảo trì. Linh hoạt

MỀM VĨNH ANH 3.4 R110



# TIẾN HOÁ PHẦN MỀM

## ① Khái niệm

Ngay sau khi phát hành lần đầu, phần mềm phải luôn không ngừng được thay đổi, phát triển, cập nhật để đáp ứng như cầu thị trường theo thời gian. Quá trình này được gọi là tiến hoá phần mềm.

## ② Khi nào phần mềm được tiến hoá?

Trong vòng đời của phần mềm, nếu không tiến hoá thì PM sẽ mất dần tính hữu dụng, vậy khi nào cần thay đổi?

- Lợi phát sinh trong quá trình vận hành, hoặc vấn đề an ninh
- Môi trường thay đổi, bao gồm phần cứng và các hệ thống tương tác, PM cần tương thích
- Yêu cầu thay đổi, theo theo PM cần đáp ứng các yêu cầu đó.
- Nghề vụ thay đổi, như cầu con người ngày càng cao và phức tạp
- Công nghệ mới phát triển

Những yếu tố trên thường xuyên xảy ra, nên PM phải tiến hoá mới có thể cạnh tranh được cũng như thu được lợi nhuận. Tuy nhiên đi kèm theo sau cũng có những khó khăn thách thức.

## ③ VTK

### SOFTWARE ENGINEERING

< Software cần phải được Engineered >

Software Engineering là qui tắc, chuẩn mực cho quá trình xây dựng phần mềm và phát triển chúng; từ những ý niệm ban đầu cho đến khi vận hành và bảo trì, SE trải dài mọi khía cạnh của sản xuất phần mềm. Về cơ bản, đây cũng là một lĩnh vực của Engineering. Trong khi Engineering nói chung, là sử dụng khoa học - toán học (lý thuyết), công cụ, phương pháp để phát minh, giải quyết vấn đề,...; thì Software Engineering cũng là đem những qui tắc (engineering discipline) ứng dụng vào quá trình phát triển thực tế của phần mềm.



# CÁC PHA (GIAI ĐOẠN) CHÍNH CỦA QUÁ TRÌNH PHÁT TRIỂN PHẦN MỀM

Lập kế hoạch dự án

Lên kế hoạch về ý định, những vấn đề đã biết, rủi ro cũng như cơ hội...  
Kèm theo những yêu cầu, chúng được ghi trong SRS. Phải có kế hoạch thi  
mô phỏng rằng đội ngũ đang đi đúng hướng.

Requirements & Analysis

BA tập hợp các yêu cầu từ khách hàng, thị trường mục tiêu, các chuyên  
giáo tạo nên Business Specification doc (hoặc CRS)...

Design

Architects và Developers tập trung thiết kế ý tưởng, lưu ý các  
yêu cầu đã đề ra và chúng sẽ ở trong D.DS.

Chủ chốt

Implementation

Devs dùng tools và ngôn ngữ lập trình để code, tạo  
SP dựa theo Design.

Testing

Đội ngũ Dev kiểm thử lỗi, thiếu sót và PM có đáp  
ứng như câu đã đề ra hay không

Deployment

Sau tất cả các bước trên, phần mềm  
được đưa đến cho khách hàng.

Maintenance

Bảo trì, nâng cấp và sửa lỗi PM,  
để không ngừng tiến hóa.

## VÒNG ĐỜI CỦA PHẦN MỀM

SDLC, là các hoạt động (quá trình) từ đầu đến cuối, từ khi lên kế hoạch cho  
đến khi phần mềm bị loại bỏ. Có thể tóm lược thành các giai đoạn chính:

Xử lý yêu cầu

Thiết kế

Kiểm thử

Vận hành - Bảo trì

# CÁC GIAI ĐOẠN PHÁT TRIỂN

