

CÔNG NGHỆ PHẦN MỀM

Khoa Công nghệ Thông tin – Đại học Đà Lạt





CHƯƠNG 1. TỔNG QUAN VỀ CÔNG NGHỆ PHẦN MỀM

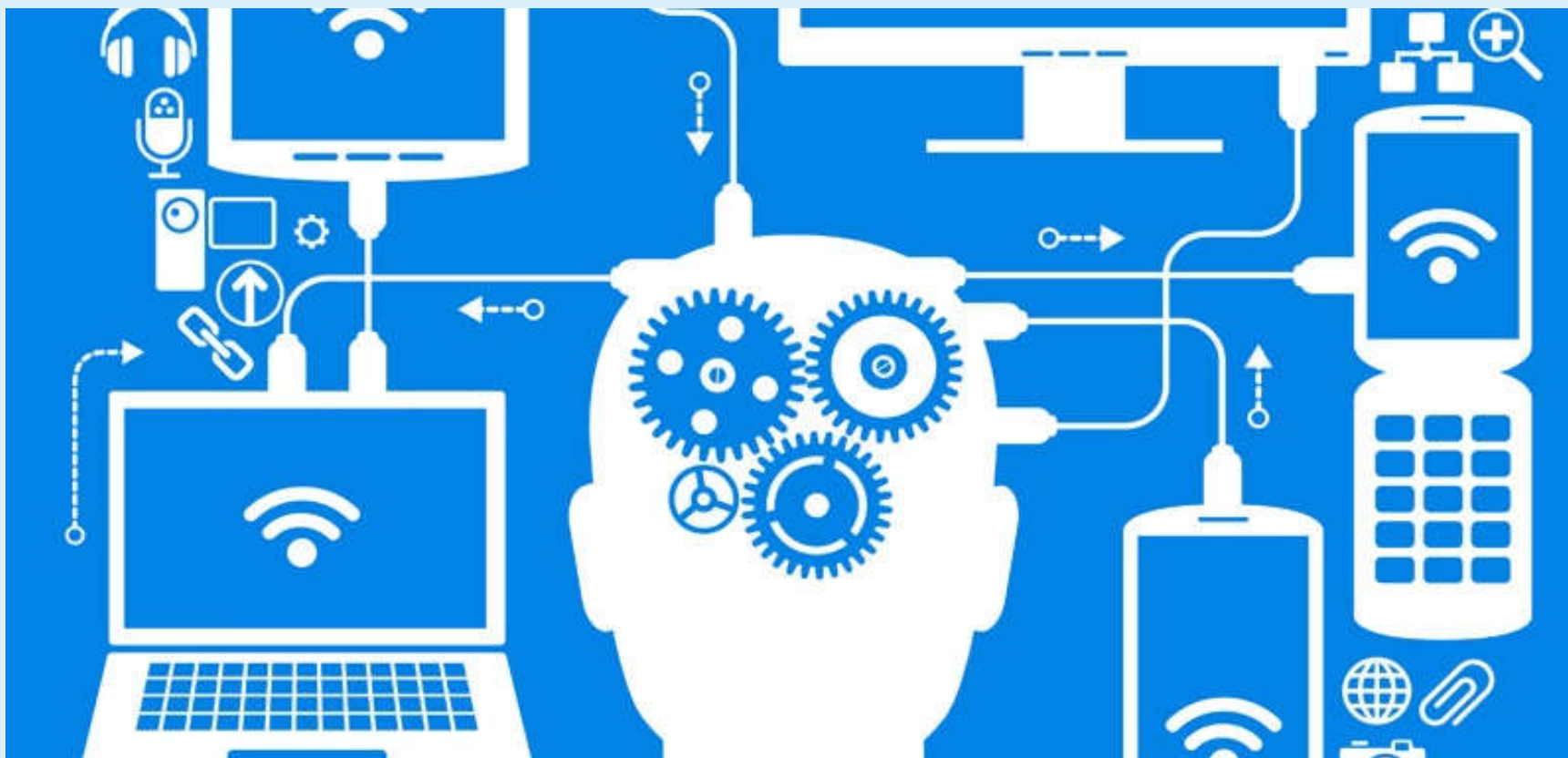
Mục tiêu

- Sau khi hoàn thành bài học này, sinh viên cần:
 - Nắm được các khái niệm: phần mềm, công nghệ phần mềm
 - Phân biệt được phần mềm và chương trình
 - Hiểu rõ cấu trúc của một phần mềm
 - Liệt kê được các giai đoạn trong quá trình phát triển phần mềm
 - Hiểu được vai trò của từng giai đoạn
 - Liệt kê được một số công cụ hỗ trợ việc phát triển phần mềm

Nội dung

- Giới thiệu
- Một số khái niệm
- Các giai đoạn phát triển phần mềm
- Các phương pháp phát triển phần mềm
- Các công cụ hỗ trợ

Giới thiệu



Phần mềm là gì?

- Sản phẩm do các chuyên gia phần mềm xây dựng và hỗ trợ một thời gian dài sau đó.
- Phần mềm bao gồm:
 - Các lệnh (chương trình máy tính): khi được thực thi sẽ cung cấp các chức năng, tính năng và hiệu suất mong muốn
 - Cấu trúc dữ liệu: cho phép chương trình lưu trữ đầy đủ và thao tác với thông tin
 - Tài liệu: mô tả cách vận hành và sử dụng chương trình



Software

=



instructions (program)

+



data structures

+



documents

Phân loại sản phẩm phần mềm

- Phần mềm đại chúng (generic products)
 - Được các lập trình viên phát triển để tung ra thị trường ở dạng miễn phí hoặc có bản quyền.
 - Chạy độc lập
 - Ví dụ: phần mềm soạn thảo, chương trình đồ họa, công cụ quản lý dự án, hệ thống đặt hẹn khám chữa bệnh, ...
- Phần mềm theo đơn hàng (customized/bespoke products)
 - Được đặt hàng để phát triển cho một khách hàng riêng lẻ theo một yêu cầu chuyên dụng.
 - Ví dụ: các hệ thống nhúng, hệ thống điều khiển không lưu hay giao thông, hệ thống rút tiền tự động (ATM), ...

Tầm quan trọng của phần mềm

- Nền kinh tế của mọi quốc gia phát triển đều phụ thuộc vào phần mềm.
- Ngày càng có nhiều hệ thống được điều khiển bởi phần mềm: giao thông, y tế, truyền thông, quân sự, công nghiệp, giải trí, ...
- Kỹ nghệ phần mềm liên quan tới nghiên cứu lý thuyết, các phương pháp và công cụ để phát triển các phần mềm chuyên nghiệp.
- Chi phí bỏ ra cho phần mềm chiếm một tỉ lệ đáng kể trong GNP của mọi quốc gia phát triển.
- Công sức của con người giảm đáng kể khi công việc được điều khiển một cách tự động bởi phần mềm trong khi hiệu suất tăng đáng kể

Các chi phí cho phần mềm

- Chi phí cho phần mềm thường chiếm phần lớn trong hệ thống máy tính.
- Chi phí cho phần mềm trên một máy tính PC thường lớn hơn chi phí cho phần cứng.
- Chi phí bảo trì phần mềm còn lớn hơn chi phí để phát triển ra nó. Với những hệ thống có thời gian tồn tại lâu dài, chi phí bảo trì có khi gấp vài lần chi phí phát triển.
- Công nghệ phần mềm liên quan tới việc phát triển các phần mềm một cách có hiệu quả về chi phí.

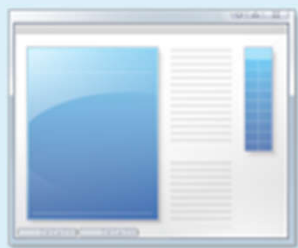
Các đặc trưng của phần mềm

- Đặc trưng: thể hiện phần mềm khác biệt với những thứ khác mà con người xây dựng.
- Gồm có:
 - Phần mềm được phát triển (develop) hay thiết kế (engineer), không phải được sản xuất theo nghĩa cổ điển, trong đó có vấn đề về chất lượng.
 - Phần mềm không có “hao mòn” (wear out) nhưng nó bị suy giảm (deteriorate) do thay đổi.
 - Mặc dù ngành công nghiệp đang hướng tới việc xây dựng phần mềm dựa trên các thành phần (component) nhưng hầu hết các phần mềm vẫn được phát triển theo dạng tùy chỉnh được. Thành phần có thể tái sử dụng sẽ đóng gói dữ liệu và xử lý vào từng phần của phần mềm để có thể sử dụng lại cho các chương trình khác.
 - Ví dụ: giao diện người dùng, cửa sổ, menu, ...

Phân loại phần mềm

- Phần mềm hệ thống: trình biên dịch, công cụ quản lý tập tin, ...
- Phần mềm ứng dụng: những chương trình độc lập, phục vụ nhu cầu cụ thể
- Phần mềm khoa học/kỹ thuật: đặc trưng bởi các thuật toán tính toán siêu tốc, chẳng hạn như trong phân tích ứng suất (của tàu vũ trụ) tự động, sinh học phân tử, động lực học chất điểm, ...
- Dòng sản phẩm phần mềm tập trung vào một thị trường có giới hạn để giải quyết thị trường tiêu dùng đại chúng, chẳng hạn như xử lý văn bản, đồ họa, quản lý CSDL, ...
- Phần mềm nhúng: nằm trong một sản phẩm hay hệ thống khác, chẳng hạn như keypad trong lò vi sóng, chức năng kỹ thuật số hiển thị bảng điều khiển trong xe hơi, ...
- WebApps: các ứng dụng web, ứng dụng thương mại điện tử trực tuyến, ...
- Phần mềm AI: sử dụng các thuật toán phi số để giải quyết các vấn đề phức tạp như khoa học rô-bốt, hệ chuyên gia, nhận dạng mẫu, ...

Cấu trúc phần mềm



Giao diện
người dùng



Xử lý



Dữ liệu

Chất lượng phần mềm



Tính chính xác



Tính tiến hóa



Tính hiệu quả



Tính tiện dụng



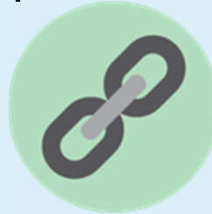
Tính tương thích



Tính tái sử dụng

Các tính chất của một phần mềm “tốt”

Tính phụ thuộc
(dependability)



Dễ bảo trì
(maintainability)



Tính bảo mật
(security)

Tính chấp nhận
(acceptability)



Tính hiệu quả
(efficiency)

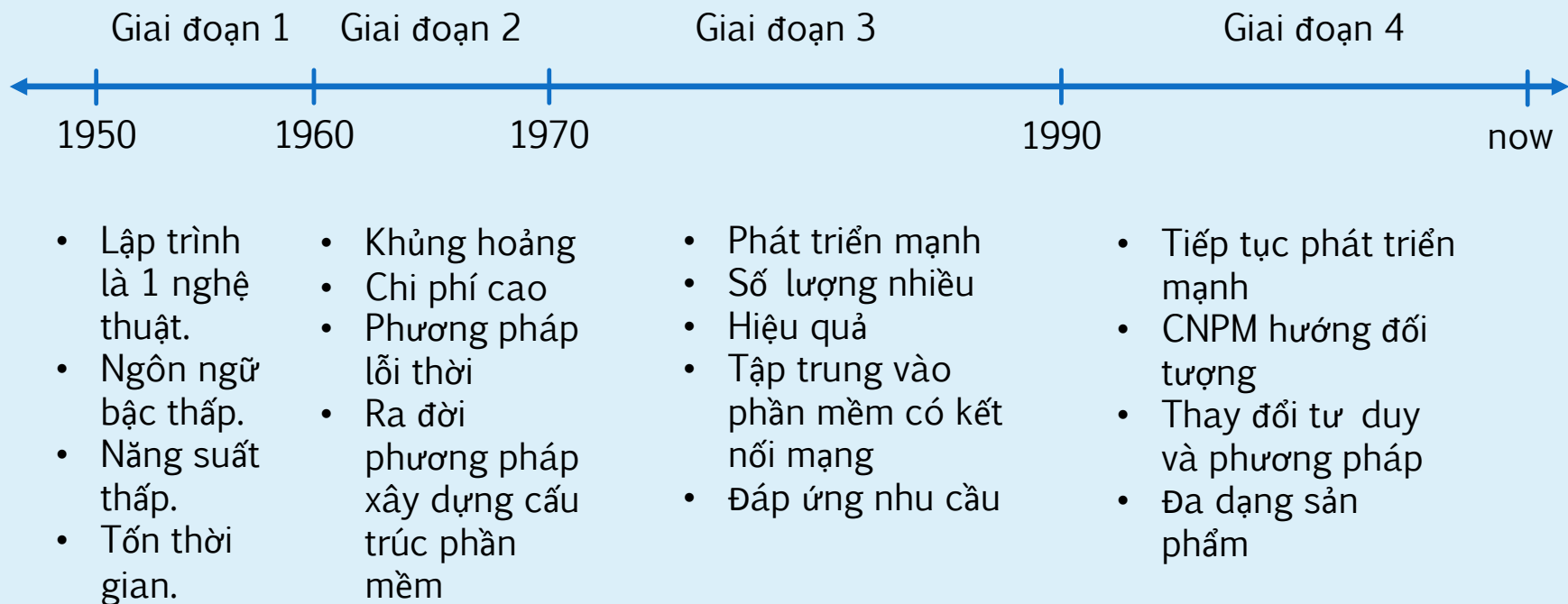
Công nghệ phần mềm

- Seminal
 - CNPM là việc thiết lập và sử dụng các nguyên tắc kỹ thuật để tạo ra phần mềm có giá trị kinh tế, hoạt động hiệu quả và đáng tin cậy trên các máy thật.
- IEEE
 - CNPM là (1) việc áp dụng cách tiếp cận có tính hệ thống, có nguyên tắc và có tính định lượng để phát triển, vận hành và bảo trì phần mềm, hay nói khác hơn là ứng dụng kỹ thuật vào phần mềm và (2) nghiên cứu các hướng tiếp cận như trong (1).

Tầm quan trọng của Công nghệ phần mềm

- Con người và xã hội ngày càng dựa vào các hệ thống phần mềm cao cấp. Vì vậy, cần phải có khả năng tạo ra các hệ thống đáng tin cậy, có lợi ích kinh tế và nhanh chóng.
- Về lâu dài, việc áp dụng các phương pháp và kỹ thuật công nghệ phần mềm cho các hệ thống phần mềm sẽ tiết kiệm hơn. Với hầu hết các hệ thống, phần lớn chi phí được đầu tư cho việc thay đổi phần mềm sau khi nó đã được đưa vào sử dụng.
- Nếu không có CNPM:
 - Có thể không thu thập đầy đủ yêu cầu hoặc không chi tiết
 - Thiếu các phương pháp để định lượng
 - Thiếu phương pháp có hệ thống dẫn tới kiến trúc quá phức tạp hoặc yếu
 - Giao diện người dùng không nhất quán
 - Mã nguồn phức tạp, khó bảo trì
 - Không có chiến lược kiểm thử có tính hệ thống dẫn đến việc xảy ra lỗi khắp nơi và kéo dài

Sơ lược về lịch sử CNPM



Các giai đoạn phát triển phần mềm

- Xác định yêu cầu
- Phân tích hệ thống
- Thiết kế
- Cài đặt (xây dựng, thực thi)
- Kiểm thử
- Triển khai
- Bảo trì

Xác định yêu cầu

- Hình thành bài toán hoặc đề tài
- Khảo sát hệ thống
- Đặt ra những câu hỏi với người sử dụng hệ thống để biết các nghiệp vụ cơ bản
- Xác định các vai trò của phần mềm
- Ước lượng công việc và lập lịch biểu phân công công việc cho nhân viên
- Kết quả: Tài liệu khảo sát yêu cầu

Phân tích hệ thống

- Đặc tả các yêu cầu
 - Nghiệp vụ
 - Hệ thống (phần mềm)
- Gồm 2 khía cạnh
 - Tĩnh: phân tích cấu trúc, chỉ ra các đối tượng và quan hệ giữa chúng
 - Động: phân tích sự tương tác động giữa các đối tượng
- Kết quả: Tài liệu đặc tả về hệ thống

Thiết kế hệ thống

- Mô hình hóa hệ thống dựa trên các mặt:
 - Dữ liệu
 - Giao diện
 - Xử lý

Cài đặt (thực thi)

- Dùng các ngôn ngữ thích hợp để cài đặt các thao tác
- Thường áp dụng các phương pháp lập trình hiện nay như:
 - Lập trình với mô hình đa tầng (n-tier)
 - Lập trình tự sinh mã một phần
 - Lập trình hướng dịch vụ
 - ...

Kiểm thử

- Nhằm chứng minh tính đúng đắn của chương trình sau khi đã tiến hành cài đặt.
- Giúp kiểm tra tính ổn định của chương trình, tính hiệu quả
- Phân loại: Kiểm thử hộp trắng, kiểm thử hộp đen
- Các mức kiểm thử:
 - Unit test
 - Integrated test
 - Use cases test
 - ...

Triển khai

- Triển khai phần mềm ra thực tế
 - cài đặt chương trình và
 - đào tạo hướng dẫn sử dụng
- Kiểm thử Beta: người sử dụng dùng thử nhằm mục đích tìm lỗi cần thiết

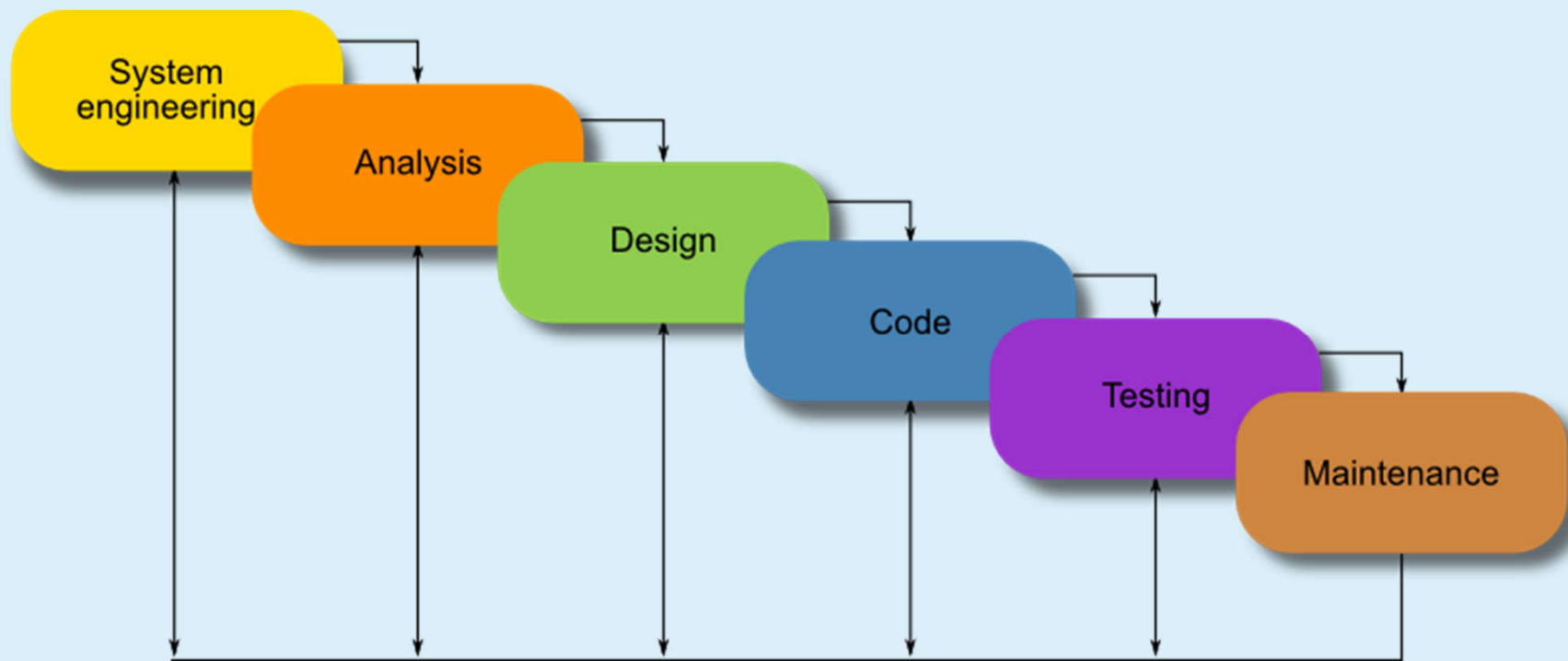
Bảo trì

- Công việc bảo trì phần mềm được tiến hành bởi một giai đoạn sau quá trình triển khai nhằm khắc phục các lỗi cần thiết để đáp ứng các yêu cầu người dùng.
- Có những chức năng sau khi triển khai mới phát sinh lỗi và nhiệm vụ của giai đoạn bảo trì là khắc phục các lỗi này.

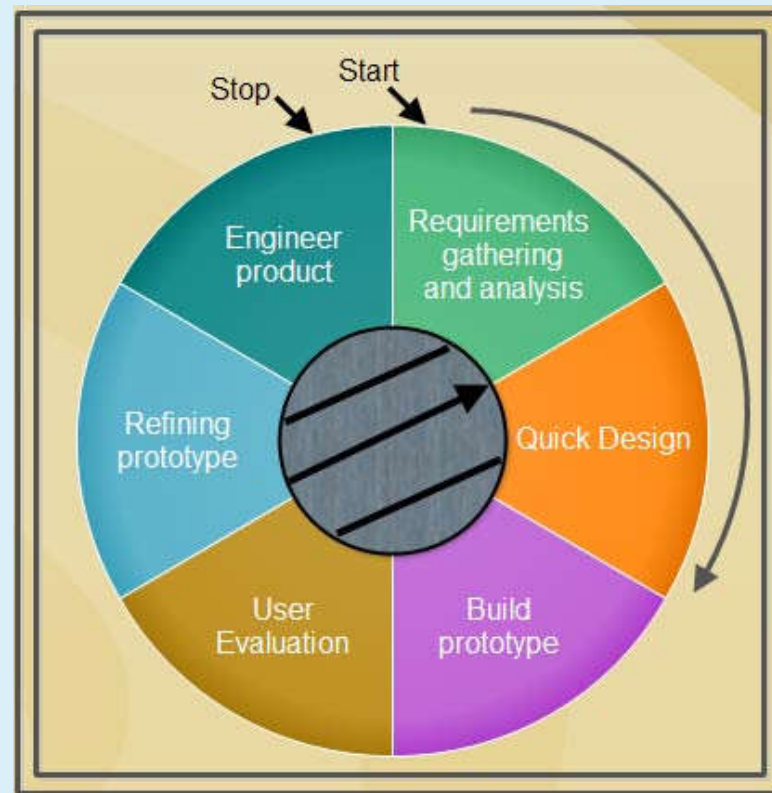
Các phương pháp phát triển phần mềm

- Mô hình thác nước
- Mô hình bản mẫu
- Mô hình xoắn ốc

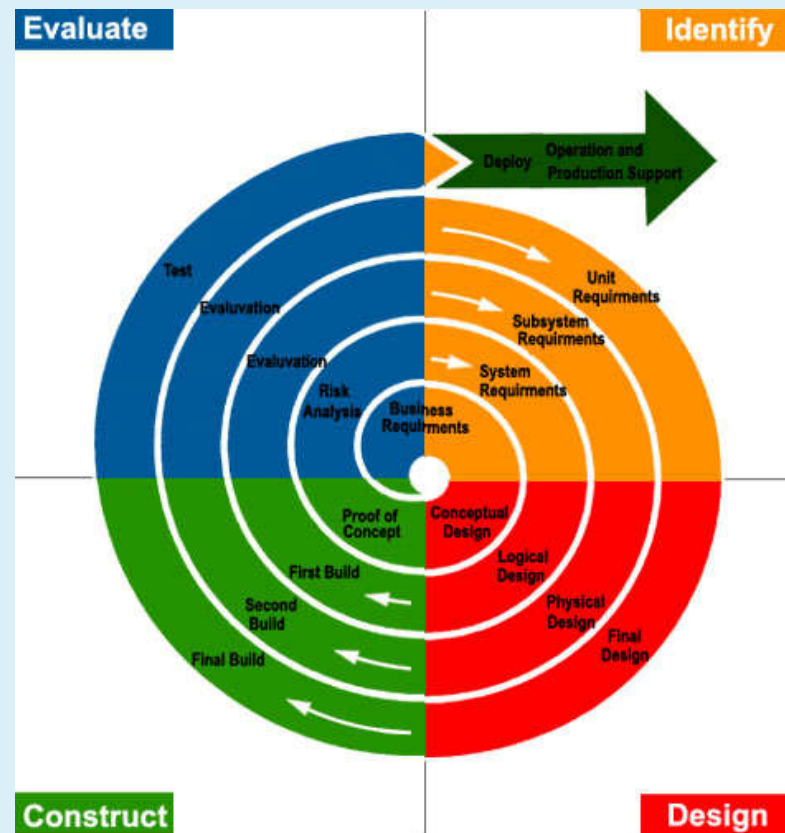
Mô hình thác nước



Mô hình bản mẫu



Mô hình xoắn ốc



Các công cụ hỗ trợ

- Phần mềm hỗ trợ phân tích: WinA&D, Analyst Pro, ...
- Phần mềm hỗ trợ thiết kế: QuickUML, Power Designer, ...
- Phần mềm hỗ trợ lập trình: Visual Studio, Net Beans, Eclipse, ...
- Phần mềm hỗ trợ kiểm thử: Ncrunch, Xmind, Nunit, ...
- Phần mềm hỗ trợ cấu hình, quản lý việc triển khai: MS Project, Redmind, Trello, MS Deployment Toolkit, Jenkins, Ansible, ...

Kết chương

- Một số kiến thức cơ bản cần nắm:
 - Các khái niệm phần mềm & công nghệ phần mềm
 - Các giai đoạn trong quá trình phát triển
 - Các công cụ để xây dựng phần mềm.

THẢO LUẬN





Question?