

Kỹ thuật phần mềm

Các quy trình kỹ nghệ yêu cầu

Khó!

- Một khách hàng vào văn phòng của bạn, ngồi xuống, nhìn thẳng vào mặt bạn và nói:



“Tôi biết anh cho là anh hiểu những gì tôi đã nói, nhưng anh không hiểu rằng những gì tôi đã nói không phải cái tôi định nói”

Mục tiêu

- Mô tả các **hoạt động** (activity) chính của kỹ nghệ yêu cầu và mối **quan hệ** giữa chúng
- Giới thiệu các kỹ thuật cho việc **thu thập** (elicitation) và **phân tích** yêu cầu
- Mô tả việc **thẩm định** (validation) yêu cầu và vai trò của việc **duyệt lại** (review) các yêu cầu
- Bàn về vai trò của **quản lý** yêu cầu trong việc hỗ trợ các quy trình kỹ nghệ yêu cầu khác

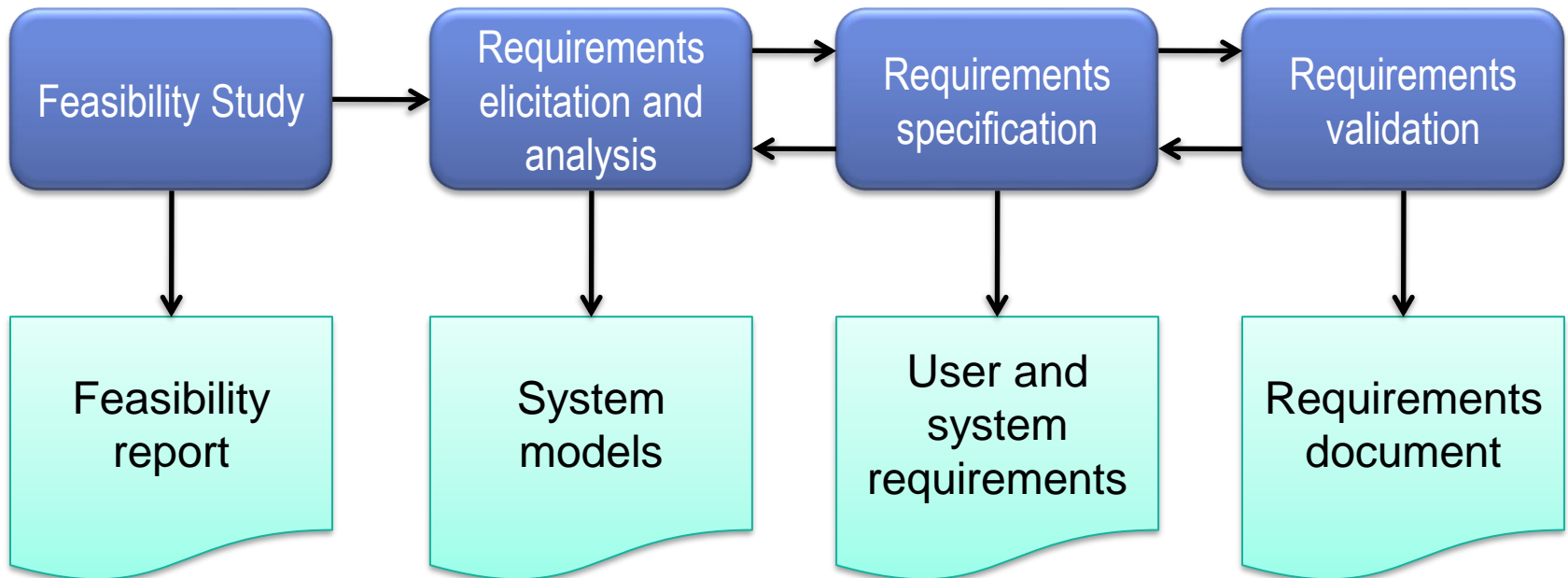
Các chủ đề

- Nghiên cứu khả thi – Feasibility studies
- Thu thập và phân tích yêu cầu – Requirements elicitation and analysis
- Thẩm định yêu cầu – Requirements validation
- Quản lý yêu cầu – Requirements management

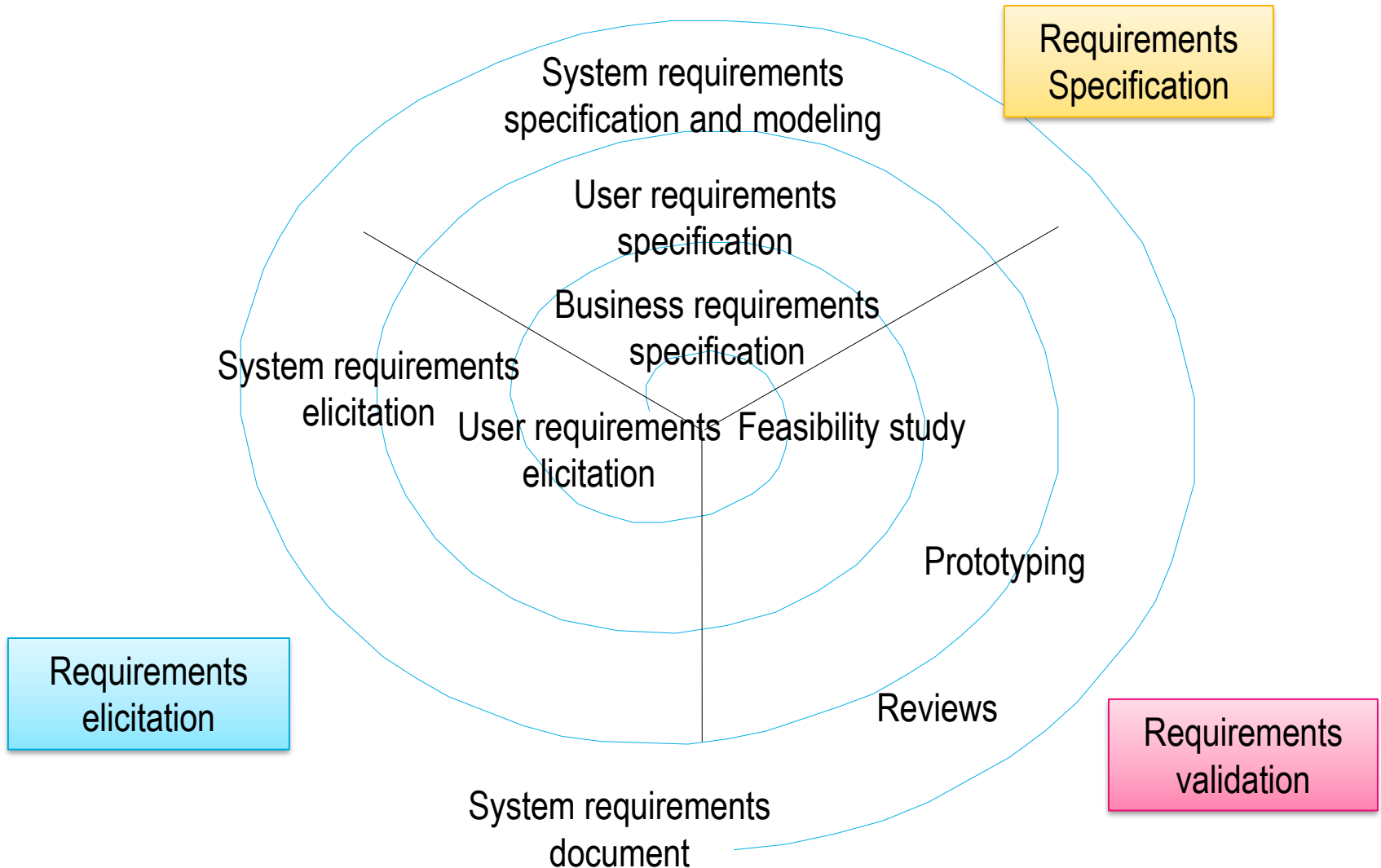
Các quy trình kĩ nghệ yêu cầu

- Rất đa dạng, tùy theo
 - Miền ứng dụng,
 - Những người có liên quan
 - Cơ quan tổ chức viết yêu cầu
- Các hoạt động tổng quát cho tất cả các quy trình
 - Requirements elicitation – thu thập
 - Requirements analysis – phân tích
 - Requirements validation – thẩm định
 - Requirements management – quản lý

The requirements engineering process



Kĩ nghệ yêu cầu



Nghiên cứu khả thi

Feasibility studies

- Một nghiên cứu ngắn, tập trung, nhằm kiểm tra xem
 - Hệ thống có đóng góp cho các mục tiêu của tổ chức hay không?
 - Hệ thống có thể được phát triển bằng công nghệ hiện hành và trong phạm vi ngân sách hay không?
 - Hệ thống có thể được tích hợp với các hệ thống khác đang được sử dụng hay không?

Thực hiện nghiên cứu khả thi

- Dựa trên đánh giá thông tin (cái gì cần), thu thập thông tin và viết báo cáo.
- Các câu hỏi dành cho nhân viên của tổ chức
 - Nếu hệ thống không được cài đặt thì sao?
 - Quy trình hiện hành có những vấn đề gì?
 - Hệ thống được đề xuất sẽ giúp được gì và như thế nào?
 - Khi tích hợp sẽ gặp những rắc rối nào?
 - Có cần công nghệ mới hay không? Cần kĩ năng gì?
 - Hệ thống mới cần hỗ trợ những tiện ích nào?

Thu thập và phân tích

- Các nhân viên kĩ thuật làm việc với khách hàng để tìm hiểu thông tin về
 - Miền ứng dụng,
 - Các dịch vụ mà hệ thống cần cung cấp và
 - Các ràng buộc về vận hành hệ thống.
- Những người có thể cần tham gia: người sử dụng, quản lý, kĩ sư bảo trì, chuyên gia miền,...
 - stakeholders.

Khó khăn khi phân tích yêu cầu

- Stakeholder không biết họ thực sự muốn gì.
- Stakeholder diễn đạt yêu cầu bằng các thuật ngữ của họ.
- Các stakeholder khác nhau có thể có các yêu cầu xung đột.
- Các nhân tố tổ chức và **chính trị** có thể ảnh hưởng đến yêu cầu hệ thống.
- Các yêu cầu **thay đổi** ngay trong quá trình phân tích
 - Chẳng hạn khi môi trường doanh nghiệp thay đổi.

Vòng xoắn ốc yêu cầu



Các hoạt động quy trình

- Discovery – Phát hiện
 - Tương tác với các stakeholder để tìm ra yêu cầu của họ.
 - Các domain requirement cũng được phát hiện tại bước này.
- Classification and organisation – Phân loại và tổ chức
 - Phân nhóm các yêu cầu có liên quan đến nhau và tổ chức chúng thành các cụm có quan hệ gắn kết với nhau.
- Prioritisation and negotiation – đặt thứ tự ưu tiên và giải quyết mâu thuẫn giữa các yêu cầu
 - Xếp thứ tự ưu tiên cho các yêu cầu và giải quyết các xung đột/mâu thuẫn giữa các yêu cầu.
- Documentation – viết tài liệu
 - Ghi lại các yêu cầu làm tài liệu đầu vào cho vòng xoắn tiếp theo.

Phát hiện yêu cầu

- Quy trình **thu thập** thông tin về hệ thống đề xuất và các hệ thống sẵn có, **gạn lọc** ra các yêu cầu người dùng và yêu cầu hệ thống từ các thông tin này.
- Các nguồn thông tin gồm có **tài liệu**, **stakeholder** hệ thống, và đặc tả của các **hệ thống tương tự**.

Các phương pháp để phát hiện yêu cầu

1. Phỏng vấn
2. Quan sát
3. Điều tra bằng bảng hỏi
4. Nghiên cứu tài liệu
5. Joint Application Design – JAD
6. Làm bản mẫu
7. Mô hình hóa/Dùng ký pháp đồ hoạ

Ví dụ: ATM stakeholder



- Khách hàng của ngân hàng (người sử dụng dịch vụ)
- Đại diện của các ngân hàng khác (ATM của ngân hàng này có thể dùng để giao dịch với ngân hàng khác)
- Quản lý ngân hàng (dùng thông tin quản lý từ hệ thống ATM)
- Nhân viên tại các chi nhánh ngân hàng (vận hành hệ thống)
- Quản trị cơ sở dữ liệu (tích hợp hệ thống với CSDL của ngân hàng)
- Quản lý an ninh
- Phòng marketing (muốn dùng ATM để quảng cáo)
- Kỹ sư bảo trì phần mềm và phần cứng
- Những người điều phối hệ thống ngân hàng quốc gia (đảm bảo hệ thống tuân theo nguyên tắc chung)

Viewpoint

- Viewpoint là một cách cấu trúc các yêu cầu để thể hiện các góc nhìn của các stakeholder khác nhau.
 - Stakeholder có thể được phân loại theo các viewpoint của họ.
- Kiểu phân tích đa diện này rất quan trọng vì không có một cách đúng duy nhất cho việc phân tích yêu cầu hệ thống.

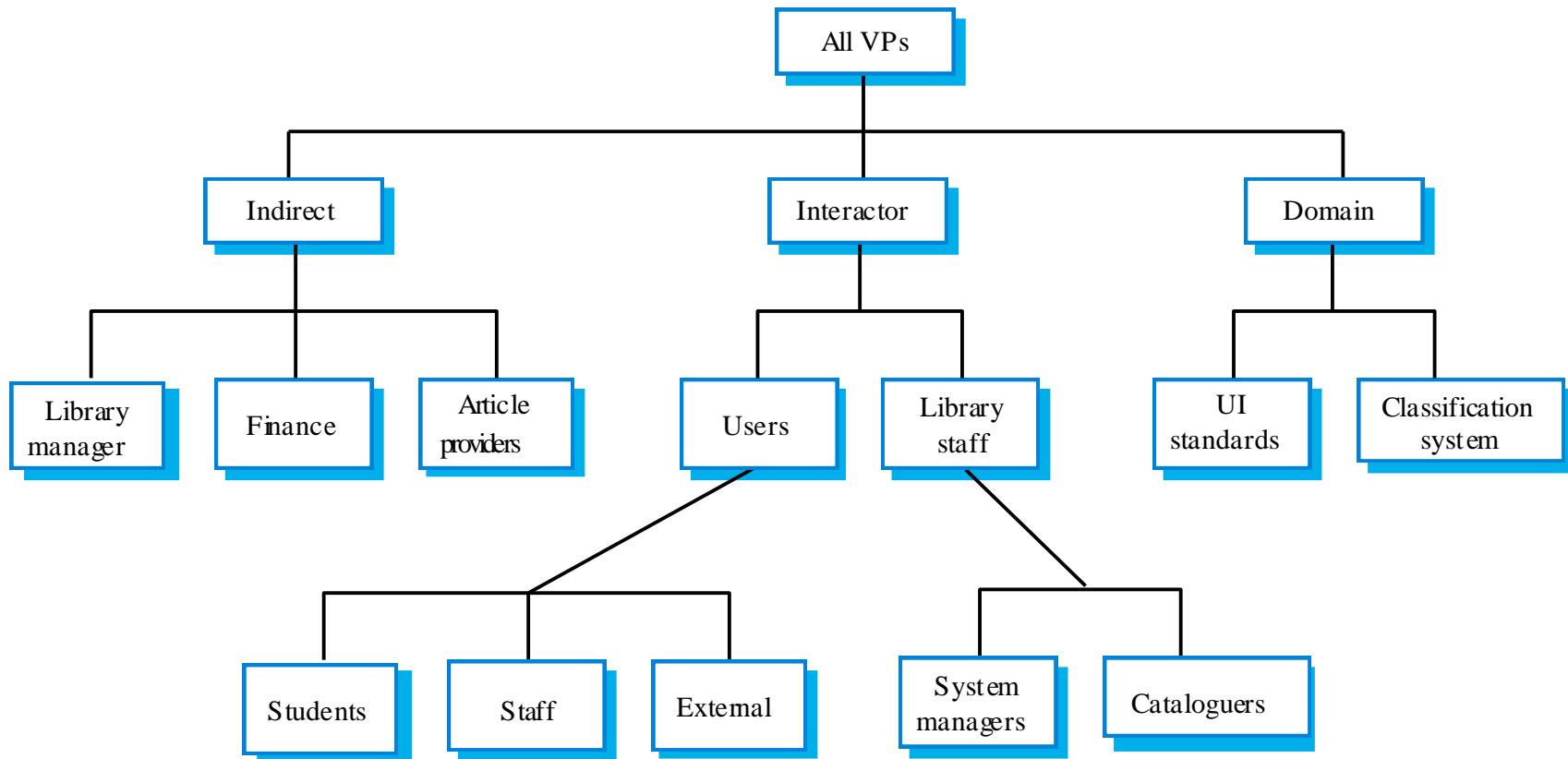
Các kiểu viewpoint

- Viewpoint tương tác - interactor viewpoint
 - Những người hoặc hệ thống khác có tương tác trực tiếp với hệ thống.
 - ở ví dụ ATM, khách hàng và CSDL tài khoản là các viewpoint tương tác.
- View point gián tiếp – Indirect viewpoint
 - Những stakeholder không trực tiếp sử dụng hệ thống nhưng có ảnh hưởng đối với các yêu cầu.
 - Quản lý và nhân viên an ninh trong ví dụ ATM
- View point miền – Domain viewpoint
 - Các đặc điểm ràng buộc trong lĩnh vực miền ứng dụng có ảnh hưởng đến các yêu cầu.
 - ở ví dụ ATM là chuẩn liên lạc giữa các ngân hàng.

Nhận diện các viewpoint

- Nhận diện các viewpoint dựa theo
 - Bên cung cấp và bên nhận các dịch vụ hệ thống;
 - Các hệ thống tương tác trực tiếp với hệ thống đang được đặc tả;
 - Các quy định và tiêu chuẩn;
 - Nguồn của các yêu cầu business và yêu cầu phi chức năng.
 - Các kĩ sư đã phát triển và bảo trì hệ thống;
 - viewpoint marketing và các viewpoint business khác.

Ví dụ: Phân cấp các viewpoint



Phỏng vấn

- Đội kĩ nghệ yêu cầu đặt các **câu hỏi** cho các stakeholder về hệ thống mà họ sử dụng và hệ thống cần phát triển.
- Có hai loại phỏng vấn
 - **Phỏng vấn đóng** trong đó người được phỏng vấn trả lời một tập các câu hỏi đã định sẵn.
 - **Phỏng vấn mở** trong đó không có lịch trình định sẵn mà người hỏi cùng với stakeholders khám phá các chủ đề.

Phỏng vấn trong thực tiễn

- Thường là kết hợp của cả phỏng vấn đóng và phỏng vấn mở.
- Có ích cho việc tìm hiểu tổng quan về công việc của stakeholder và họ có thể tương tác với hệ thống như thế nào.
- **Không** tốt cho việc tìm hiểu về **domain** requirement
 - Các kĩ sư thu thập yêu cầu không thể hiểu các thuật ngữ chuyên ngành;
 - Một số kiến thức chuyên ngành quá quen thuộc đối với stakeholder đến mức họ không thể nghĩ là cần phải giải thích chúng.

Làm thế nào để hỏi cho hiệu quả?

- Người hỏi cần có tư duy mở, sẵn sàng nghe stakeholder nói và không giữ các quan niệm đã có từ trước về các yêu cầu.
- Để có hiệu quả, người hỏi
 - Nên gợi ý người được phỏng vấn bằng một câu hỏi hoặc một đề xuất
 - Không nên chỉ đợi người kia trả lời những câu hỏi kiểu như ‘ông muốn gì’.

Kịch bản

- Kịch bản (scenario) là các ví dụ đời thực về việc hệ thống có thể được sử dụng như thế nào.
- Các kịch bản nên bao gồm
 - Một miêu tả về tình huống ban đầu
 - Một miêu tả về luồng sự kiện thông thường
 - Một miêu tả về những trục trặc gì có thể xảy ra
 - Thông tin về các hoạt động xảy ra đồng thời
 - Một miêu tả về trạng thái khi kịch bản kết thúc

Kịch bản (1)

Initial Assumption: Người dùng đã đăng nhập hệ thống và đã tìm thấy tạp chí có đăng tài liệu cần tìm.

Normal:

- Người dùng chọn tài liệu cần copy. Hệ thống sẽ yêu cầu người dùng nhập thông tin thuê bao hoặc chọn cách trả phí dùng tài liệu. Có thể thanh toán bằng thẻ tín dụng hoặc dùng số tài khoản của một tổ chức.
- Sau đó người dùng được yêu cầu điền một form bản quyền trong đó có chi tiết về giao dịch này, rồi submit form đó cho hệ thống LIBSYS.
- Hệ thống kiểm tra form bản quyền, nếu OK, bản PDF của tài liệu sẽ được tải xuống máy tính của người dùng và người dùng được thông báo về việc này. Sau đó người dùng được chọn một máy in, và tài liệu sẽ được in tại đó. Nếu tài liệu đã được gắn cờ 'print-only' thì nó sẽ được xóa khỏi máy của người dùng ngay sau khi người dùng khẳng định rằng đã in xong.

Kịch bản (2)

Điểm gì làm sai:

- Người dùng có thể điền form sai. Khi đó hệ thống cần hiện lại form để người dùng sửa lại. Nếu form được submit sau đó vẫn sai thì hủy yêu cầu đọc tài liệu của người dùng.
- Hệ thống có thể không chấp nhận giao dịch thanh toán tiền. Hủy yêu cầu đọc tài liệu của người dùng.
- Việc download tài liệu có thể thất bại. Làm lại cho đến khi thành công hoặc khi người dùng chấm dứt phiên làm việc.
- Có thể không in được tài liệu. Nếu bài báo không có gắn cờ 'print-only' thì giữ nó trong workspace của LIBSYS. Nếu không, xóa tài liệu và hoàn lại chi phí cho người dùng.

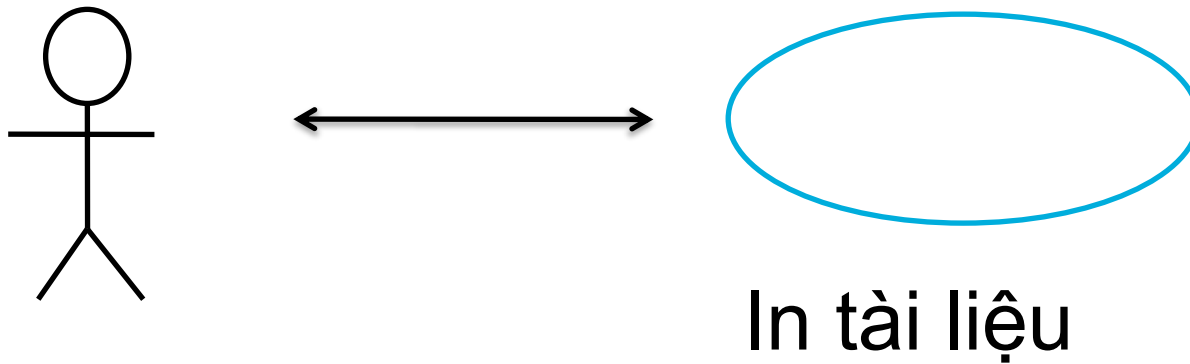
Hoạt động khác: Song song download các tài liệu khác nhau.

Trạng thái hệ thống: Người dùng đang ở trạng thái đăng nhập. Nếu tài liệu có gắn cờ 'print-only' thì nó đã bị xóa khỏi workspace.

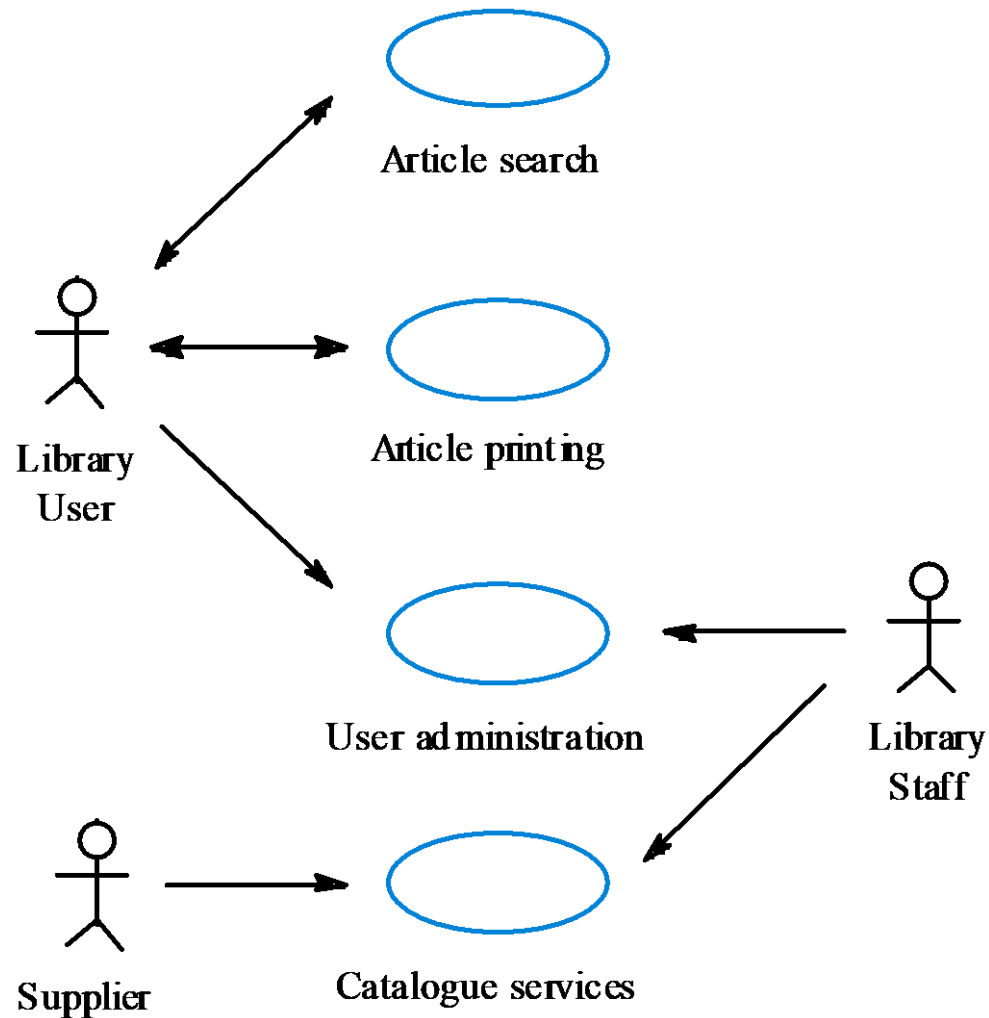
Use case

- Ca sử dụng (use-case) là một kĩ thuật kiểu kịch bản bằng ngôn ngữ UML
 - Chỉ rõ các actor trong một tương tác và
 - Mô tả chính tương tác đó.
- Một bộ ca sử dụng có thể mô tả được tất cả các tương tác có thể đối với hệ thống.
- Có thể dùng các sơ đồ tuần tự (sequence diagram) để bổ sung chi tiết cho các ca sử dụng
 - Minh họa chuỗi xử lý sự kiện.

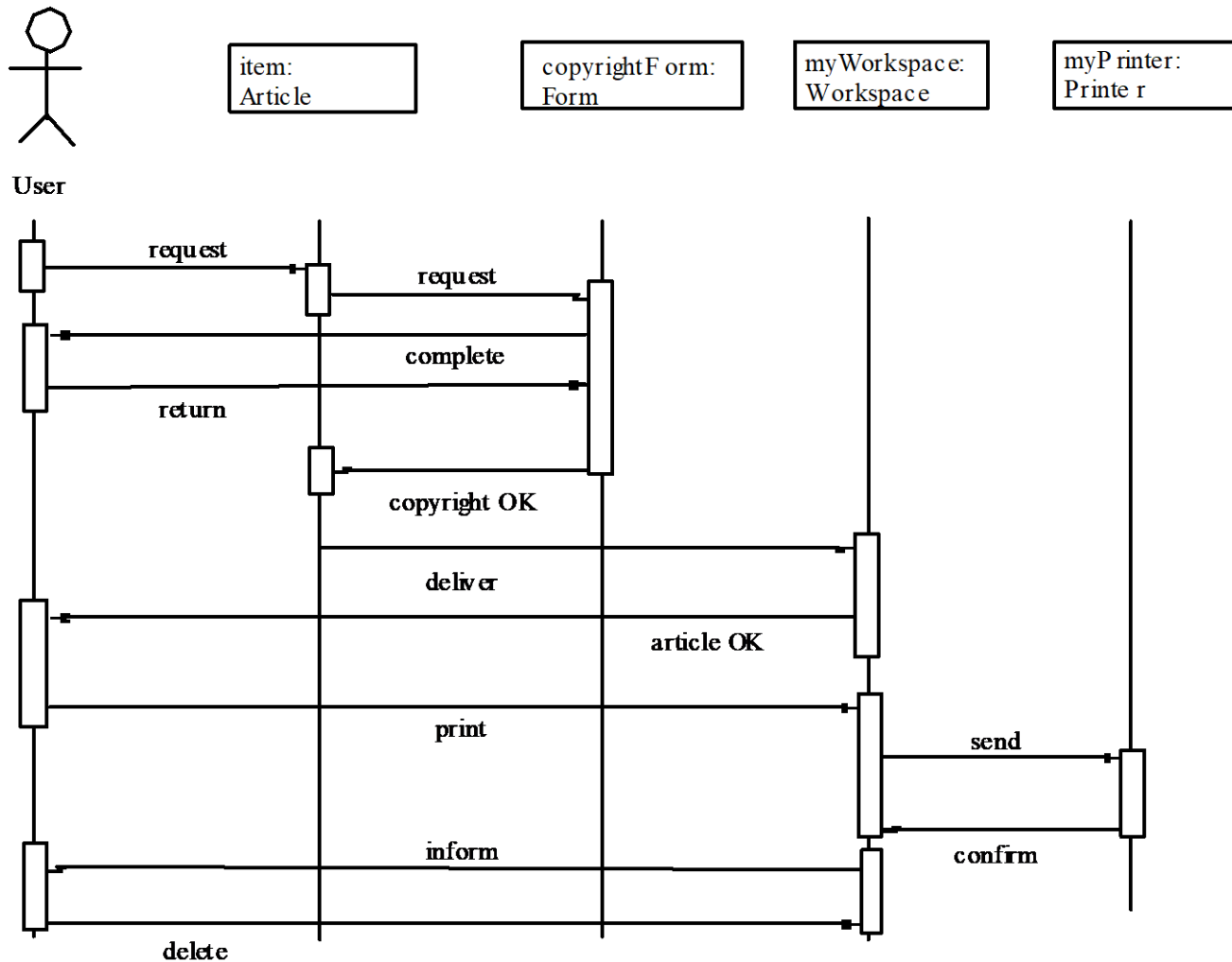
use-case in tài liệu



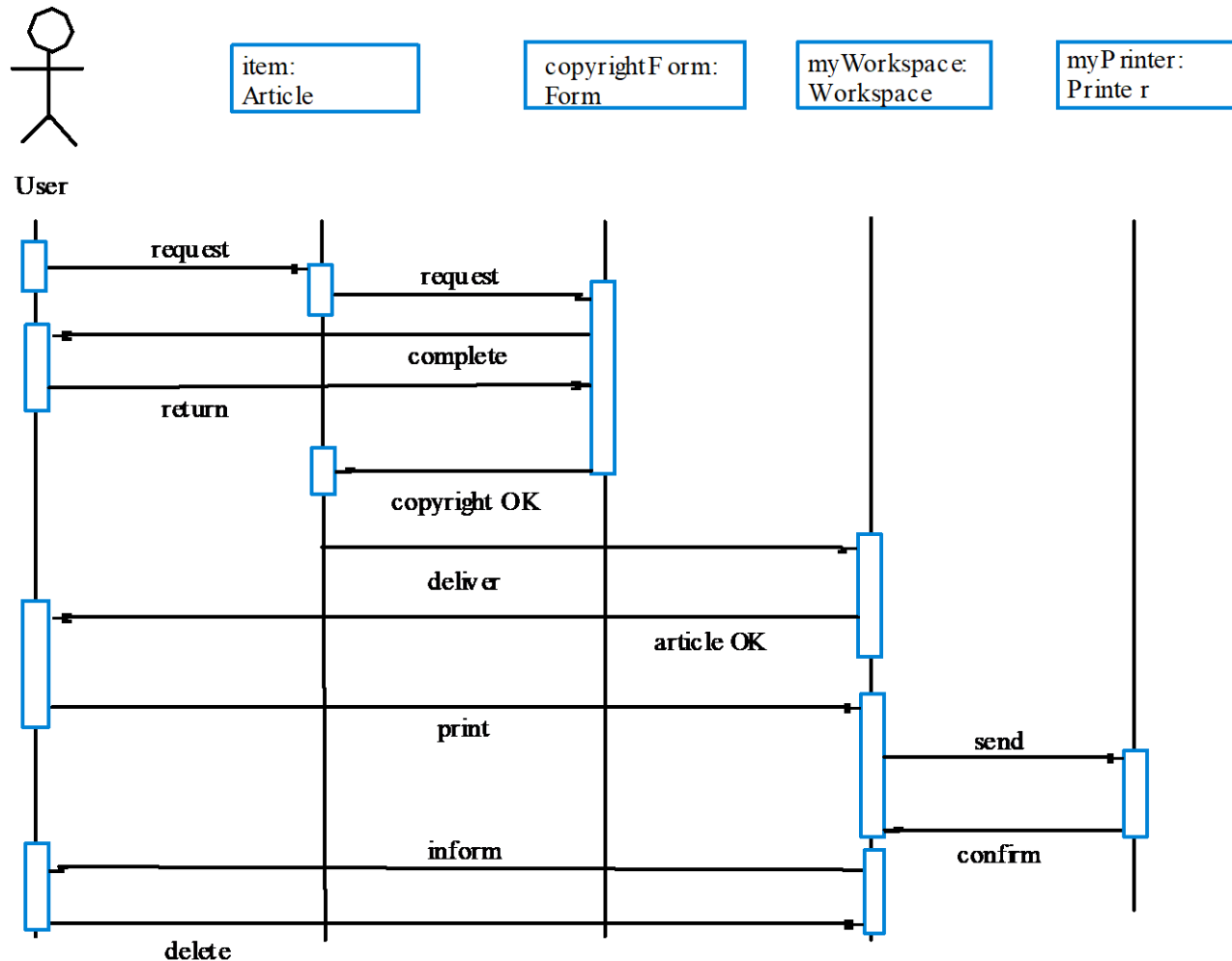
LIBSYS use case



Article printing



Print article sequence



Các nhân tố xã hội và tổ chức

- Các hệ thống phần mềm được dùng trong một ngữ cảnh tổ chức và xã hội.
 - Ngữ cảnh ảnh hưởng đến các yêu cầu hệ thống.
- Các nhân tố xã hội và tổ chức không phải một viewpoint đơn nhất mà ảnh hưởng đến **tất cả** các viewpoint.
- Những nhà phân tích giỏi phải nhạy cảm với các nhân tố này
 - Nhưng hiện nay không có một phương pháp có hệ thống nào 😞

Thẩm định yêu cầu

- (Requirement validation) quan tâm đến việc chứng tỏ rằng các yêu cầu định nghĩa được hệ thống mà **khách hàng thực sự muốn**.
- Chi phí của lỗi yêu cầu cao đến mức công đoạn thẩm định rất quan trọng
 - Việc sửa một lỗi yêu cầu sau khi đã bàn giao phần mềm có thể tốn kém gấp 100 lần chi phí cho việc sửa một lỗi cài đặt.

Kiểm tra yêu cầu

- Hiệu lực – Validity
 - Hệ thống có cung cấp những chức năng phục vụ tốt nhất yêu cầu của khách hàng hay không?
- Nhất quán – Consistency
 - Có những yêu cầu nào xung đột nhau không?
- Đầy đủ – Completeness
 - Có đủ các chức năng mà khách hàng đòi hỏi hay không?
- Thực tế – Realism
 - Có thể cài đặt các yêu cầu trong phạm vi công nghệ và ngân sách cho phép hay không?
- Kiểm định được – Verifiability
 - Có cách kiểm tra các yêu cầu xem chúng đã được thỏa mãn chưa hay không?

Kĩ thuật thẩm định yêu cầu

- Duyệt yêu cầu – Requirements reviews
 - Đọc và phân tích lại một cách có hệ thống (không dùng chương trình tự động).
- Phiên bản thử nghiệm – Prototyping
 - Dùng một mô hình chạy được của hệ thống để kiểm tra các yêu cầu (Chương 17)
- Sinh test-case – Test-case generation
 - Phát triển các test dành cho các yêu cầu để kiểm tra khả năng kiểm thử được.

Review yêu cầu

- Nên đều đặn tổ chức các buổi review trong giai đoạn định nghĩa yêu cầu đang được hình thành.
- Nhân viên của cả bên A và bên B cần tham gia review.
- Các cuộc review có thể chính thức (với các tài liệu hoàn chỉnh) hoặc không chính thức.
 - Giao tiếp tốt giữa đội phát triển, khách hàng, và người sử dụng có thể giúp giải quyết các vấn đề ngay từ giai đoạn đầu.

Cần kiểm tra cái gì khi review?

- **Kiểm định được – Verifiability**
 - Về thực tiễn, có test được yêu cầu không?
- **Dễ hiểu – Comprehensibility**
 - Yêu cầu có được hiểu đúng hay không?
- **Lần vết được – Traceability**
 - Nguồn gốc của yêu cầu có được ghi rõ hay không? Đôi khi cần lần lại nguồn gốc của một yêu cầu để đánh giá ảnh hưởng của một thay đổi
- **Thích nghi được – Adaptability**
 - Yêu cầu có thể thay đổi mà không gây ảnh hưởng lớn tới các yêu cầu khác hay không?

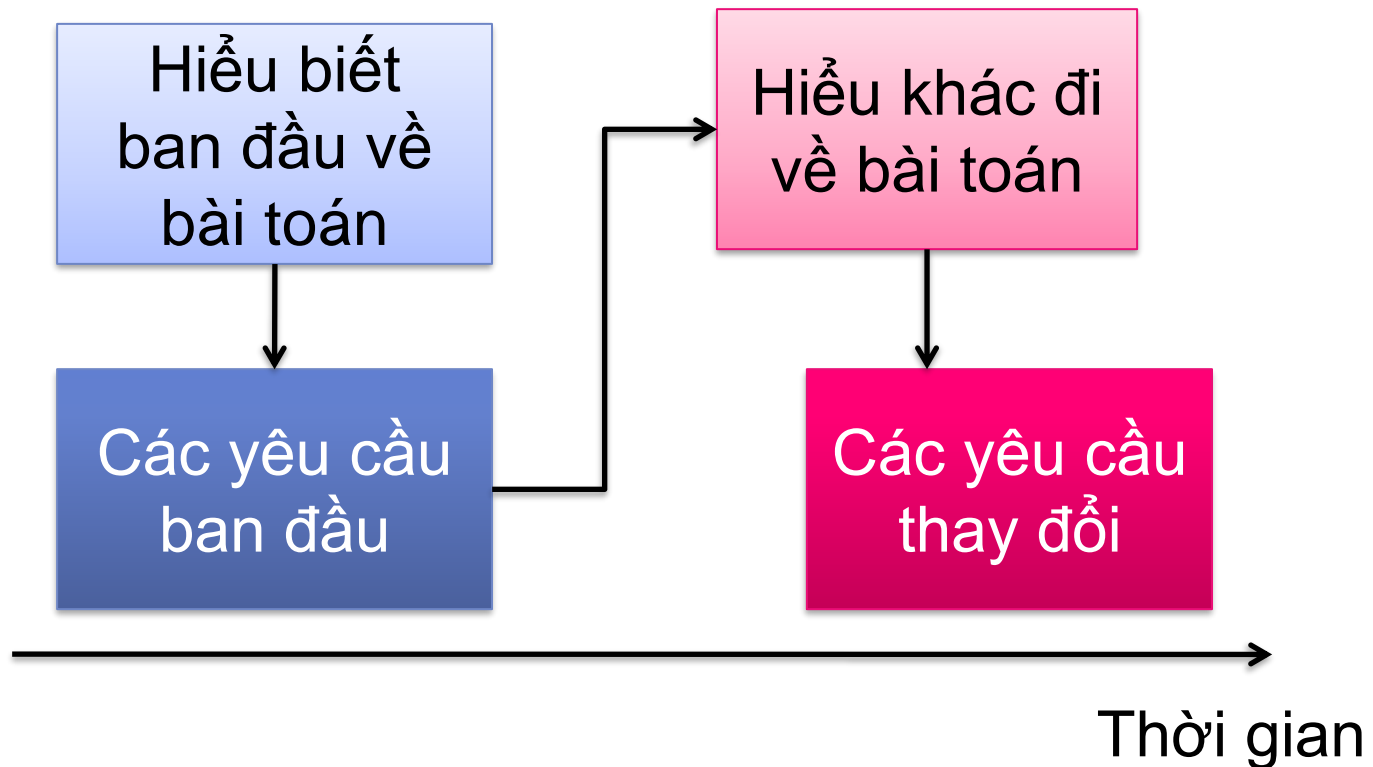
Quản lý yêu cầu

- Quy trình quản lý các yêu cầu thay đổi
 - Trong quá trình kỹ nghệ yêu cầu và phát triển hệ thống.
- Bộ các yêu cầu thường không đầy đủ và không nhất quán, điều đó không thể tránh được
 - Các yêu cầu mới xuất hiện trong quá trình phát triển phần mềm
 - Doanh nghiệp cần thay đổi
 - Hiểu biết hơn về hệ thống đang được phát triển
 - Các viewpoint khác nhau có các yêu cầu khác nhau và chúng thường xung đột

Thay đổi về yêu cầu

- **Mức độ ưu tiên** của các yêu cầu từ các viewpoint khác nhau thay đổi trong quá trình phát triển.
- Từ góc nhìn doanh nghiệp, khách hàng hệ thống có thể đưa ra các yêu cầu **xung đột** với yêu cầu của người sử dụng.
- **Môi trường** doanh nghiệp và kĩ thuật của hệ thống thay đổi trong quá trình phát triển hệ thống.

Sự tiến hóa của yêu cầu



Các yêu cầu bền vững và dễ thay đổi

- Các yêu cầu bền vững
 - Các yêu cầu ổn định được rút ra từ hoạt động cốt lõi của tổ chức khách hàng.
 - Ví dụ, bệnh viện bao giờ cũng có bác sĩ, y tá, v.v. có thể rút ra từ các mô hình miền ứng dụng
- Các yêu cầu dễ thay đổi
 - Các yêu cầu thay đổi trong quá trình phát triển hệ thống hoặc khi hệ thống đang được sử dụng..
 - Trong ngữ cảnh bệnh viện, các yêu cầu rút ra từ chính sách chăm sóc sức khỏe

Phân loại yêu cầu

Kiểu	Miêu tả
Mutable (dễ thay đổi)	Các yêu cầu thay đổi do thay đổi môi trường hoạt động của tổ chức. Ví dụ, trong các hệ thống bệnh viện, ngân sách dành cho bệnh nhân có thể thay đổi và do đó đòi hỏi thu thập loại thông tin khác về điều trị.
Emergent (phát sinh)	Các yêu cầu nảy sinh khi khách hàng hiểu hơn về hệ thống trong khi hệ thống đang được phát triển. Quy trình thiết kế có thể làm phát sinh những yêu cầu mới.
Consequential (hệ quả)	Các yêu cầu là hệ quả của việc tin học hóa. Việc đưa hệ thống máy tính vào sử dụng có thể làm thay đổi các quy trình của tổ chức và mở ra các cách làm việc mới, từ đó sinh ra các yêu cầu hệ thống mới
Compatibility (tương thích)	Các yêu cầu phụ thuộc vào các hệ thống cụ thể hoặc các quy trình doanh nghiệp cụ thể bên trong một tổ chức. Khi các hệ thống và quy trình này thay đổi, các yêu cầu về tính tương thích có thể cũng phải thay đổi theo.

Kế hoạch quản lý yêu cầu

Trong quá trình kỹ nghệ yêu cầu, ta phải lập kế hoạch cho:

- Định danh yêu cầu – Requirements identification
 - Mỗi yêu cầu phải được đánh số khác nhau để có thể dẫn chiếu từ yêu cầu này tới yêu cầu khác nhằm phục vụ đánh giá về khả năng lần vết (traceability);
- Một quy trình quản lý thay đổi
 - Là bộ các hoạt động đánh giá ảnh hưởng và chi phí của các thay đổi.
- Các chính sách lần vết – Traceability policies
 - Các chính sách này quy định cần phải ghi lại các quan hệ nào giữa các yêu cầu và giữa yêu cầu với thiết kế hệ thống, và cần phải lưu giữ các hồ sơ đó như thế nào;
- CASE tool
 - Công cụ hỗ trợ công việc quản lý các thay đổi về yêu cầu;

Khả năng lần vết – Traceability

- Khả năng lần vết liên quan đến mối quan hệ giữa các yêu cầu, nguồn gốc của chúng và thiết kế hệ thống
- **Source** traceability – lần vết nguồn
 - Liên kết từ yêu cầu tới những stakeholder đã đề xuất các yêu cầu đó;
- **Requirements** traceability – lần vết yêu cầu
 - Liên kết giữa các yêu cầu phụ thuộc nhau;
- **Design** traceability – lần vết thiết kế
 - Liên kết từ yêu cầu đến phần thiết kế tương ứng;

Một ma trận lần vết

Requirement id	1.1	1.2	1.3	2.1	2.2	2.3	3.1	3.2
1.1		D	R					
1.2			D			D		D
1.3	R			R				
2.1			R		D			D
2.2								D
2.3		R		D				
3.1								R
3.2							R	

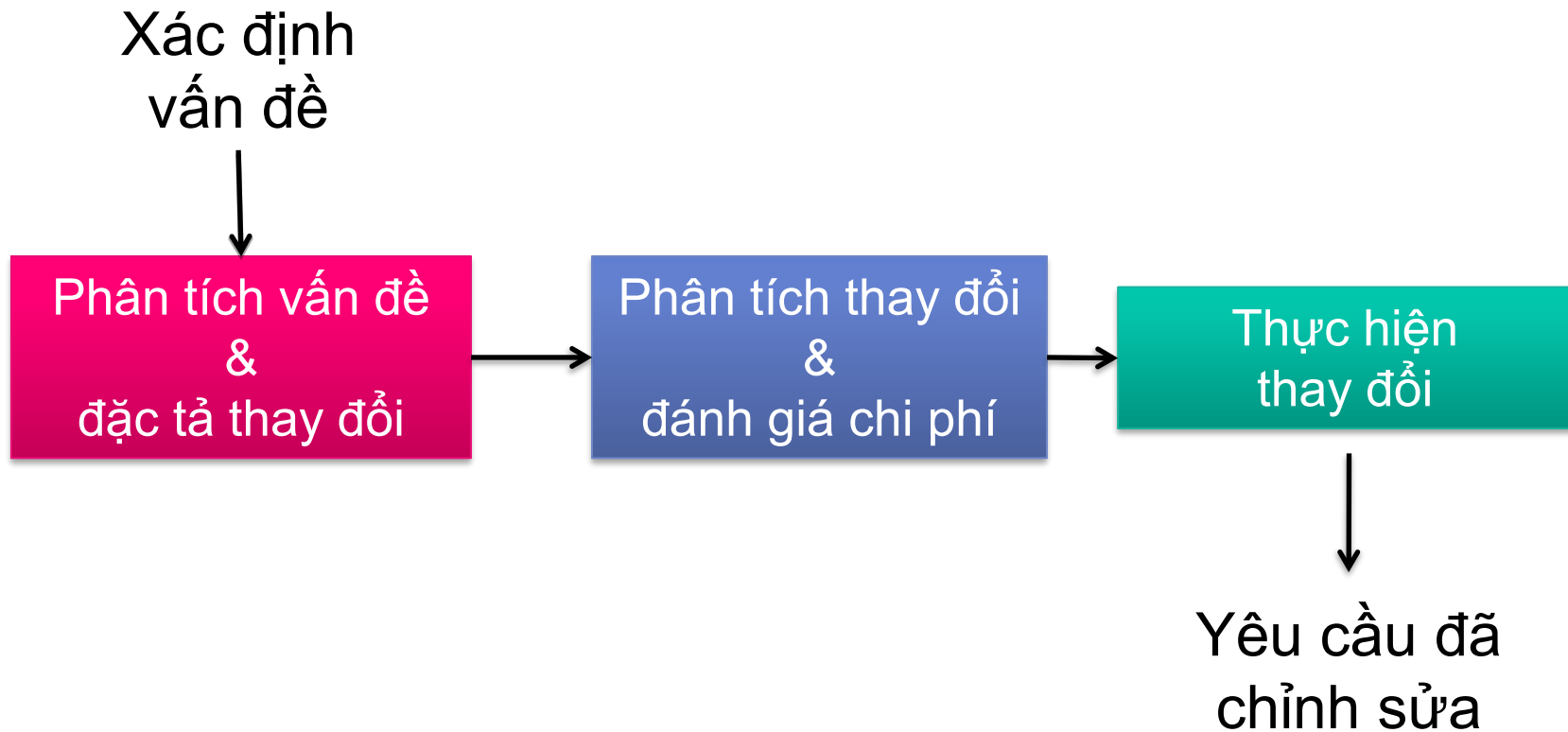
CASE tool

- Lưu trữ yêu cầu
 - Các yêu cầu cần được lưu trữ trong một kho dữ liệu an toàn và được quản lý.
- Quản lý thay đổi
 - Quy trình quản lý thay đổi là một quy trình workflow mà trong đó các giai đoạn có thể được định nghĩa và luồng thông tin chảy giữa các giai đoạn đó được tự động hóa một phần.
- Quản lý lần vết
 - Tự động tìm các liên kết giữa các yêu cầu.

Quản lý thay đổi yêu cầu

- Nên áp dụng cho tất cả các thay đổi được đề xuất đối với bộ yêu cầu.
- Các giai đoạn chính
 - Phân tích vấn đề
 - Thảo luận về vấn đề của các yêu cầu và đề xuất thay đổi;
 - Phân tích thay đổi và đánh giá chi phí.
 - Đánh giá hiệu ứng của thay đổi đối với các yêu cầu khác;
 - Thực hiện thay đổi.
 - Sửa tài liệu yêu cầu và các tài liệu khác để thực hiện thay đổi đã xét.

Quản lý thay đổi



Tổng kết

- Quy trình kĩ nghệ yêu cầu (RE process) bao gồm
 - Một nghiên cứu khả thi,
 - Thu thập và phân tích yêu cầu,
 - Thực hiện theo vòng lặp các công việc: tìm hiểu miền ứng dụng, thu thập yêu cầu, phân loại, cấu trúc, đặt mức ưu tiên và thẩm định
 - Đặc tả yêu cầu và
 - Quản lý yêu cầu.

Tổng kết

- Hệ thống có nhiều stakeholder với các yêu cầu khác nhau.
- Các nhân tố xã hội và tổ chức có ảnh hưởng đối với yêu cầu hệ thống.
- Thẩm định yêu cầu là kiểm tra tính hiệu lực, nhất quán, đầy đủ, thực tiễn và kiểm định được.
- Các thay đổi doanh nghiệp chắc chắn dẫn đến việc các yêu cầu thay đổi.
- Quản lý yêu cầu bao gồm lập kế hoạch và quản lý thay đổi.