

CÔNG NGHỆ PHẦN MỀM

SOFTWARE ENGINEERING

TS. Võ Đức Hoàng

Khoa Công nghệ thông tin

Trường Đại học Bách khoa - Đại học Đà Nẵng

- Chương 1: Giới thiệu Công nghệ phần mềm
- Chương 2: Các mô hình phát triển phần mềm
- Chương 3: Phân tích và đặc tả yêu cầu
- Chương 4: Các kỹ thuật đặc tả
- Chương 5: Thiết kế
- Chương 6: Lập trình và ngôn ngữ lập trình
- Chương 7: Kiểm thử
- Chương 8: Quản trị dự án

■ ***Giáo trình chính:***

- Andrew Troelsen, *Pro C# 5.0 and The .NET 4.5 Framework*, Apress, 2012.

■ ***Tài liệu tham khảo:***

- Simon Kendal, *Object Oriented Programming using C#*, BookBoon, 2012.
- Microsoft MSDN, *C# Programming Guide for Visual Studio 2013*.
- Joe Mayo, *LINQ Programming*, McGraw-Hill Education, 2009.
- Andrew Clymer, *Pro Asynchronous Programming with .NET*, Apress, 2013...

Phân tích và đặc tả yêu cầu (3)

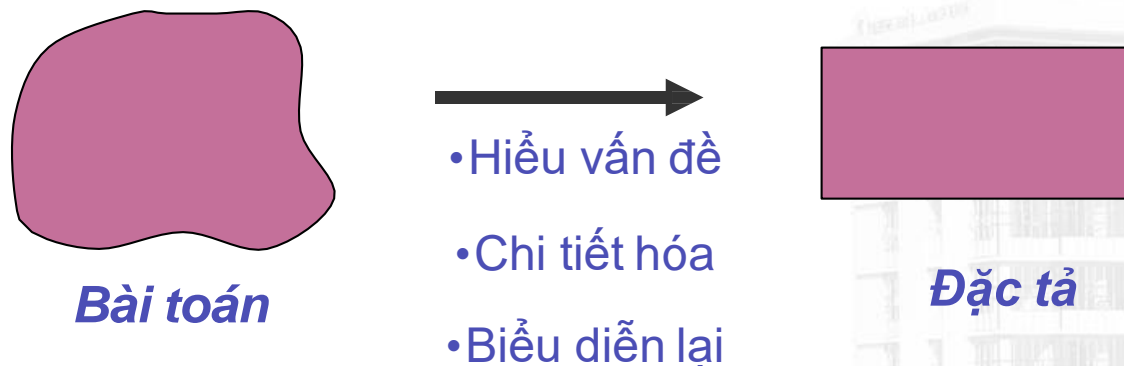
- Đại cương về phân tích và đặc tả
- Nền tảng của phân tích
- Phân tích và nắm bắt yêu cầu
- Đặc tả yêu cầu
- Thẩm định yêu cầu
- Một số phương pháp và công cụ hỗ trợ

- Phân tích và đặc tả yêu cầu là khâu kỹ thuật đầu tiên của quá trình phát triển phần mềm.
 - Xác định các chức năng/dịch vụ mà khách hàng yêu cầu từ hệ thống
 - Các ràng buộc mà hệ thống được phát triển và vận hành.
- Là sự phối hợp của nhà phát triển và khách hàng.
- Quyết định chất lượng phần mềm đạt được và chi phí bỏ ra.

- Mục đích: Đặc tả yêu cầu phần mềm
 - Cơ sở cho việc mời thầu (cần có giải thích)
 - Cơ sở để ký kết hợp đồng (cần đầy đủ chi tiết)
 - Làm tư liệu phục vụ thiết kế và triển khai (cần đầy đủ, chính xác, không mâu thuẫn)

Yêu cầu là gì?

- Các yêu cầu là các **mô tả từ mức trừu tượng rất cao đến một đặc tả rất chi tiết** về dịch vụ mà hệ thống cần cung cấp cũng như các ràng buộc lên sự phát triển và hoạt động của nó.



- Yêu cầu là:
 - Năng lực của phần mềm mà người sử dụng cần để giải quyết vấn đề đặt ra nhằm đạt được mục đích xác định
 - Năng lực của phần mềm cần có nhằm thỏa mãn một hợp đồng, một chuẩn, một đặc tả.

Yêu cầu phần mềm là gì?



- Yêu cầu phần mềm: là tất cả các yêu cầu về phần mềm do khách hàng - người sử dụng phần mềm - nêu ra,
 - Các chức năng của phần mềm,
 - Hiệu năng của phần mềm,
 - Các yêu cầu về thiết kế và giao diện
 - Các yêu cầu đặc biệt khác
- Thông thường các yêu cầu phần mềm được phân loại theo 4 thành phần của phần mềm:
 - Các yêu cầu về phần mềm (Software)
 - Các yêu cầu về phần cứng (Hardware)
 - Các yêu cầu về dữ liệu (Data)
 - Các yêu cầu về con người (People, Users)

Tại sao phải đặt ra yêu cầu phần mềm?



- Khách hàng chỉ có những ý tưởng còn mơ hồ về phần mềm cần phải xây dựng để phục vụ công việc của họ,
 - Chúng ta phải sẵn sàng, kiên trì theo đuổi để đi từ các ý tưởng mơ hồ đó đến “Phần mềm có đầy đủ các tính năng cần thiết”
- Khách hàng rất hay thay đổi các đòi hỏi của mình,
 - Chúng ta nắm bắt được các thay đổi đó và sửa đổi các mô tả một cách hợp lý

- Những vấn đề từ phía người dùng:
 - Người dùng không hiểu họ muốn gì
 - Người dùng liên tục thay đổi yêu cầu ngay cả khi việc phát triển sản phẩm đã được bắt đầu
 - Người dùng không hiểu về kỹ thuật
 - Người dùng không hiểu về quy trình phát triển

- Những vấn đề từ phía nhà phát triển:
 - Ngôn từ của người dùng và nhà phát triển không khớp nhau
 - Nhà phát triển cố lái cho yêu cầu của người dùng khớp với một hệ thống hay mô hình sẵn có thay vì phát triển một hệ thống theo nhu cầu của khách hàng
 - Việc phân tích có thể do các lập trình viên thực hiện thay vì các nhân viên có kỹ năng phân tích để có thể hiểu được nhu cầu của khách hàng một cách đúng đắn

■ Những vấn đề khác:

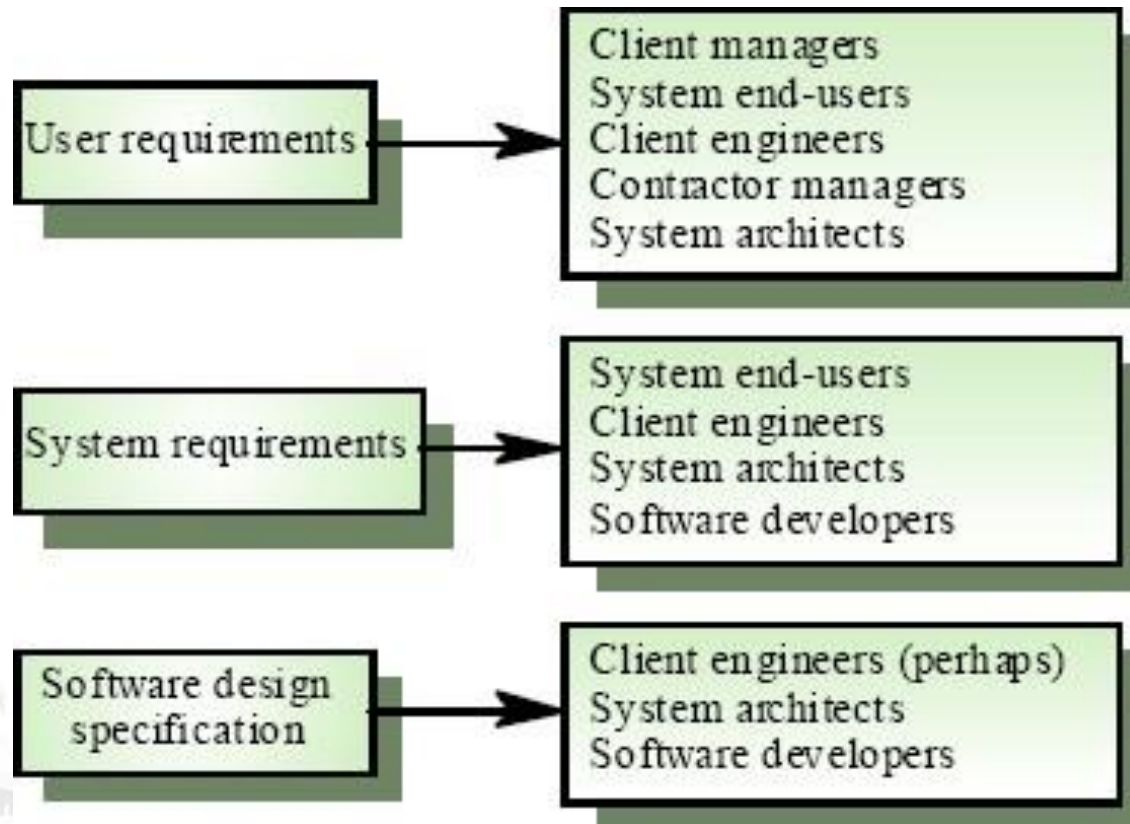
- Các yêu cầu thường mang tính đặc thù của tổ chức đặt hàng nó, do đó nó thường khó hiểu, khó định nghĩa và không theo một tiêu chuẩn nào cả
- Các hệ thống thông tin lớn có rất nhiều người sử dụng, do đó các yêu cầu thường rất đa dạng và có các mức ưu tiên khác nhau, thậm chí mâu thuẫn lẫn nhau
- Người đặt hàng nhiều khi là các nhà quản lý, không phải là người dùng thực sự do đó việc đưa ra các yêu cầu thường không chính xác

■ Yêu cầu

- Nếu một công ty muốn có một hợp đồng cho một dự án phát triển phần mềm lớn, thì họ phải xác định nhu cầu của mình theo một cách đủ trù tượng mà chưa cần xác định trước một giải pháp.
- Các yêu cầu phải được viết để một số nhà thầu có thể đấu thầu hợp đồng, cung cấp, có thể theo các cách khác nhau để đáp ứng nhu cầu của tổ chức khách hàng.
- Khi một hợp đồng đã được trao, nhà thầu phải viết một định nghĩa hệ thống cho khách hàng chi tiết hơn để khách hàng hiểu và có thể xác nhận những gì phần mềm sẽ làm. Cả hai tài liệu này có thể được gọi là tài liệu yêu cầu cho hệ thống.

- Yêu cầu người dùng
 - Diễn đạt bằng ngôn ngữ tự nhiên và sơ đồ
 - Nêu rõ dịch vụ hệ thống cung cấp và các ràng buộc trong hoạt động của nó.
- Yêu cầu hệ thống
 - Mô tả đủ chi tiết về các dịch vụ hệ thống
 - Như một hợp đồng giữa khách hàng và nhà thầu
- Đặc tả phần mềm
 - Mô tả chi tiết về phần mềm làm cơ sở cho thiết kế và cài đặt
 - Dành cho người phát triển

Đối tượng đọc yêu cầu



=> Yêu cầu viết ra cần đáp ứng được tất cả các đối tượng này.

- Các yêu cầu chức năng
 - Mô tả các chức năng hay các dịch vụ mà hệ thống phần mềm cần cung cấp.
- Các yêu cầu phi chức năng
 - Mô tả các ràng buộc đặt lên dịch vụ và quá trình phát triển hệ thống (về chất lượng, về môi trường, chuẩn sử dụng, qui trình phát triển..)
- Các yêu cầu miền/lĩnh vực
 - Những yêu cầu đặt ra từ miền ứng dụng, phản ánh đặc trưng của miền đó.

- Mô tả chức năng hay các dịch vụ của hệ thống
- Chúng phụ thuộc vào:
 - Loại phần mềm sẽ được xây dựng
 - Sự mong muốn của khách hàng
 - Loại hệ thống mà phần mềm trợ giúp
- Mức độ các yêu cầu
 - Trừu tượng: hệ thống làm gì
 - Chi tiết: dịch vụ cụ thể hệ thống thực hiện

■ Ví dụ (1):

- Người sử dụng có thể tìm kiếm tài liệu dựa trên từ khóa có trong tài liệu hoặc tên tài liệu
- Hệ thống cần cung cấp cho người sử dụng phương tiện hiển thị dễ dàng các tài liệu từ CSDL
- Hệ thống phải đọc được các định dạng khác nhau của tài liệu: text, doc, pdf, ppt, bảng tính excel,...

■ Ví dụ (2):

Các yêu cầu chức năng đối với MHC-PMS

- Người dùng sẽ có thể tìm kiếm danh sách bệnh nhân đã hẹn cho tất cả các phòng khám.
- Mỗi ngày Hệ thống sẽ tạo ra cho mỗi phòng khám một danh sách các bệnh nhân dự kiến đến đến hẹn khám vào ngày hôm đó.
- Mỗi nhân viên sử dụng hệ thống sẽ được nhận dạng duy nhất bằng mã số nhân viên gồm 8 chữ số.

- Sự không chính xác của yêu cầu
 - Yêu cầu không được phát biểu chính xác
 - Yêu cầu nhập nhằng có thể được hiểu khác nhau giữa người sử dụng và người phát triển
 - Ví dụ: yêu cầu “hiển thị dễ dàng”
 - Người sử dụng: có thể hiển thị các loại tài liệu khác nhau
 - Người phát triển: cung cấp giao diện hiển thị tài liệu ở chế độ văn bản
 - Trên nguyên tắc yêu cầu phải thỏa mãn
 - Đầy đủ: yêu cầu phải mô tả đầy đủ các chức năng cần thiết
 - Gắn bó: các yêu cầu không mâu thuẫn nhau
 - Thực tế không đơn giản để có các yêu cầu đầy đủ và gắn bó

- Ví dụ: Sự không chính xác của yêu cầu

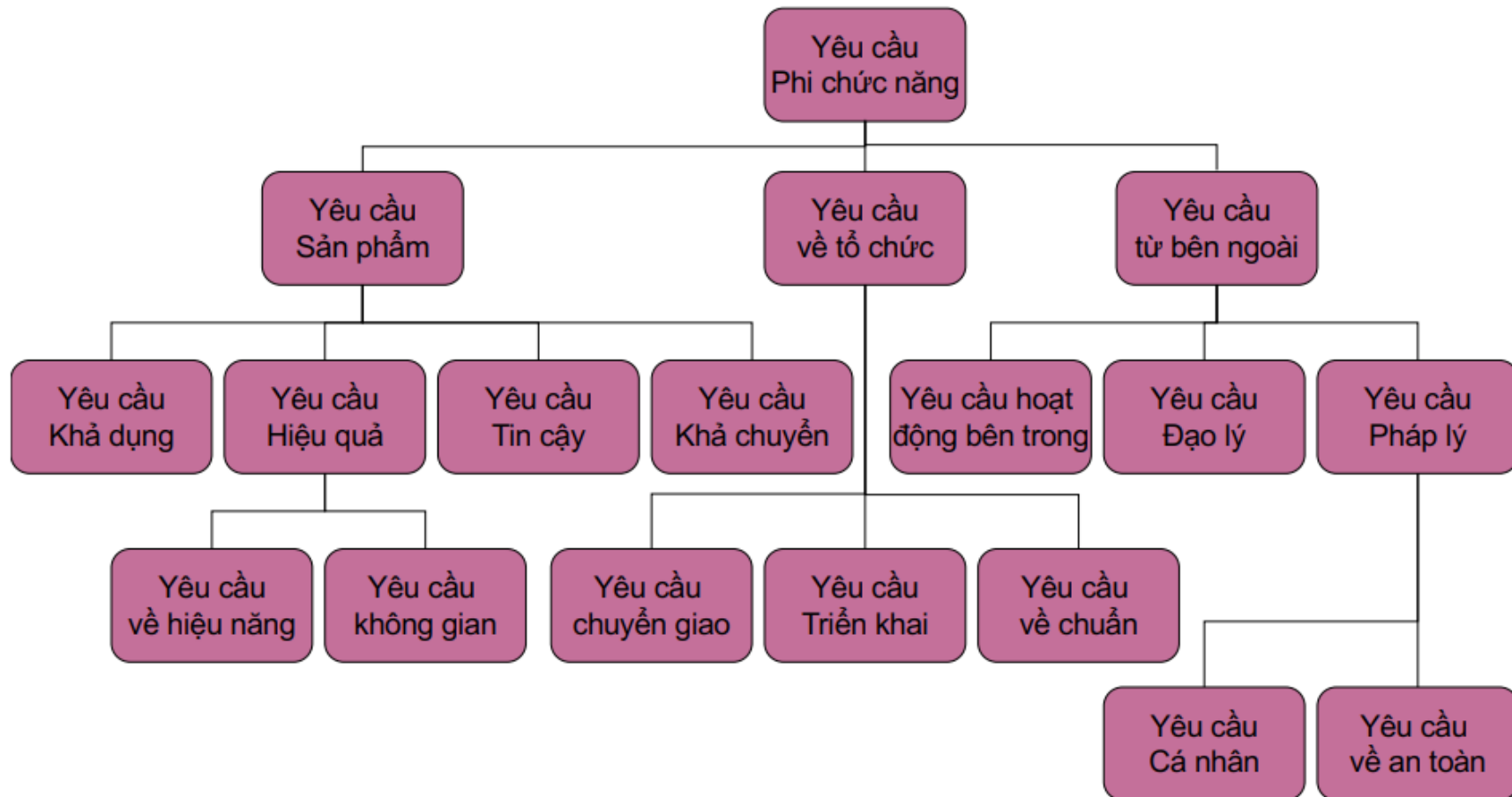
Xem xét thuật ngữ 'tìm kiếm' trong yêu cầu 1 - Yêu cầu chức năng đối với MHC-PMS

1. Mục đích của người dùng - tìm kiếm tên bệnh nhân trong tất cả các cuộc hẹn ở tất cả các phòng khám;
2. Ý kiến của nhà phát triển - tìm kiếm tên bệnh nhân trong một phòng khám cá nhân. Người dùng chọn phòng khám rồi tìm kiếm..

- Định nghĩa các tính chất và ràng buộc của hệ thống
 - Yêu cầu tiến trình
 - Phương pháp thiết kế
 - Ngôn ngữ lập trình
 - Công cụ sử dụng
 - Thời gian trả lời
 - Độ tin cậy
 - Yêu cầu về lưu trữ dữ liệu
- Yêu cầu phi chức năng có thể quan trọng hơn yêu cầu chức năng
 - Nếu yêu cầu phi chức năng không được đáp ứng hệ thống trở nên vô dụng

- Yêu cầu về sản phẩm
 - Tốc độ, độ tin cậy, bộ nhớ cần, giao diện...
- Yêu cầu về tổ chức, tiến trình phát triển
 - Các chuẩn áp dụng, phương pháp thiết kế, ngôn ngữ lập trình, mô hình tiến trình, yêu cầu cài đặt...
- Yêu cầu từ bên ngoài
 - Yêu cầu đến từ các yếu tố bên ngoài hệ thống và tiến trình phát triển: yêu cầu về khả năng tương tác, về đạo đức, ..
 - Về chi phí, về bản quyền, liên kết,...

Các loại yêu cầu phi chức năng



■ Ví dụ

- Yêu cầu về sản phẩm

- Phần mềm chỉ nên yêu cầu tối đa 256 MB bộ nhớ
- Hệ thống phải được cài đặt bằng HTML mà không có frame hoặc Java applets.
- Những người sử dụng có kinh nghiệm có thể sử dụng được tất cả các chức năng của hệ thống chỉ sau hai tiếng tập huấn.
- Sau khoá huấn luyện, số lỗi chương trình gây ra bởi người sử dụng là không quá hai lỗi một ngày.

■ Ví dụ

- Yêu cầu về tổ chức

- Phải lắp ghép hệ thống với hệ thống đang sử dụng, bàn giao việc quản lý CSDL hoàn toàn...

- Yêu cầu bên ngoài

- Hệ thống không được để lộ thông tin cá nhân của khách hàng ra bên ngoài

- Ví dụ về các yêu cầu phi chức năng trong MHC-PMS:
 - Yêu cầu về sản phẩm: MHC-PMS sẽ có sẵn cho tất cả các phòng khám trong giờ làm việc bình thường (Thứ Hai - Thứ Sáu, 08:30 - 17.30). Thời gian ngừng hoạt động trong giờ làm việc bình thường không được vượt quá năm giây trong một ngày bất kỳ.
 - Yêu cầu về tổ chức: Người sử dụng hệ thống MHC-PMS phải tự xác thực bằng chứng minh thư của cơ quan y tế.
 - Yêu cầu bên ngoài: Hệ thống phải thực hiện các điều khoản về quyền riêng tư của bệnh nhân như được nêu trong HStan-03-2006-priv.

- Các yêu cầu phi chức năng có thể rất khó để mô tả chính xác và các yêu cầu không chính xác khó có thể xác minh.
- Mục đích
 - Một ý định chung của người dùng như dễ sử dụng.
- Yêu cầu phi chức năng có thể kiểm chứng
 - Một phát biểu sử dụng một số độ đo có thể được kiểm tra khách quan.
- Mục tiêu rất hữu ích cho các nhà phát triển khi họ truyền đạt ý định của người dùng hệ thống.

■ Ví dụ: Yêu cầu về tính khả dụng

- Hệ thống phải dễ sử dụng bởi nhân viên y tế và phải được tổ chức theo cách mà lỗi của người dùng được giảm thiểu.
- Nhân viên y tế sẽ có thể sử dụng tất cả các chức năng của hệ thống sau bốn giờ đào tạo. Sau khóa đào tạo này, số lỗi trung bình do người dùng có kinh nghiệm mắc phải sẽ không vượt quá hai lỗi mỗi giờ sử dụng hệ thống. (Yêu cầu phi chức năng có thể kiểm tra)

- Miền hoạt động của hệ thống bắt buộc áp dụng các yêu cầu trên hệ thống.
 - Ví dụ, một hệ thống điều khiển tàu phải tính đến các đặc tính phanh trong các điều kiện thời tiết khác nhau.
- Yêu cầu miền là các yêu cầu chức năng mới, các ràng buộc đối với các yêu cầu hiện có hoặc xác định các tính toán cụ thể.
- Nếu các yêu cầu miền không được thỏa mãn, hệ thống có thể không hoạt động.

■ Hiểu được

- Các yêu cầu được thể hiện bằng ngôn ngữ của miền ứng dụng;
- Điều này thường không được hiểu bởi các kỹ sư phần mềm phát triển hệ thống.

■ Nhân chứng

- Các chuyên gia miền lĩnh vực hiểu rõ lĩnh vực đến mức họ không nghĩ đến việc làm rõ ràng cho các yêu cầu miền.

- Các yêu cầu đối với một hệ thống phần mềm đặt ra những gì hệ thống nên làm và xác định các ràng buộc đối với hoạt động và triển khai.
- Các yêu cầu chức năng là các phát biểu về các dịch vụ mà hệ thống phải cung cấp hoặc là các mô tả về cách thực hiện một số tính toán.
- Các yêu cầu phi chức năng thường ràng buộc hệ thống đang được phát triển và quá trình phát triển đang được sử dụng.
- Chúng thường liên quan đến các thuộc tính nổi bật của hệ thống và do đó áp dụng cho toàn bộ hệ thống.

TÀI LIỆU YÊU CẦU

- Nên mô tả
 - Yêu cầu chức năng
 - Yêu cầu phi chức năng
- Dễ hiểu đối với người sử dụng
 - Không có kiến thức chi tiết về kỹ thuật/tin học
- Nên mô tả
 - Bằng ngôn ngữ tự nhiên
 - Biểu đồ, bảng biểu

■ Ưu điểm

- Dễ hiểu
- Dễ sử dụng

■ Hạn chế

- Không rõ ràng, thiếu chính xác
- Nhập nhằng
- Lẫn lộn giữa yêu cầu chức năng và yêu cầu phi chức năng
- Quá mềm dẻo
 - Có thể trình bày nhiều cách

- Thiếu rõ ràng
 - Độ chính xác là khó khăn mà không làm cho tài liệu khó đọc.
- Yêu cầu nhầm lẫn
 - Các yêu cầu chức năng và phi chức năng có xu hướng lẫn lộn.
- Yêu cầu hợp nhất
 - Một số yêu cầu khác nhau có thể được thể hiện cùng nhau.

Giải pháp thay thế cho NNTN



- Ngôn ngữ có cấu trúc
 - Sử dụng ngôn ngữ gần gũi với ngôn ngữ lập trình
- Các mô hình
 - Các ký hiệu đồ họa
- Ký hiệu toán học
 - Ngôn ngữ hình thức

- Là đặc tả chi tiết hơn yêu cầu người sử dụng
- Phục vụ cơ bản cho bước thiết kế
- Có thể sử dụng làm một phần của hợp đồng
- Có thể sử dụng các mô hình để mô tả

- Tài liệu yêu cầu phần mềm là các phát biểu chính thức về những gì được yêu cầu của người phát triển phần mềm
- Nên bao gồm cả định nghĩa yêu cầu người dùng và đặc tả yêu cầu hệ thống
- Không phải là tài liệu thiết kế. Nó là tập hợp những gì hệ thống nên làm thay vì là làm như thế nào.

- Chỉ mô tả về **chức năng** (các hành vi bên ngoài) của hệ thống và các **ràng buộc (WHAT)**
- Không mô tả về phương thức cài đặt (**HOW**) – không phải là tài liệu thiết kế
- Phải dễ thay đổi
 - Khó xác định được đầy đủ, chính xác ngay
 - Phải qua nhiều bước xét duyệt lại
- Sử dụng như là tài liệu tham khảo khi bảo trì
- Đặc tả trả lời các sự kiện không mong đợi

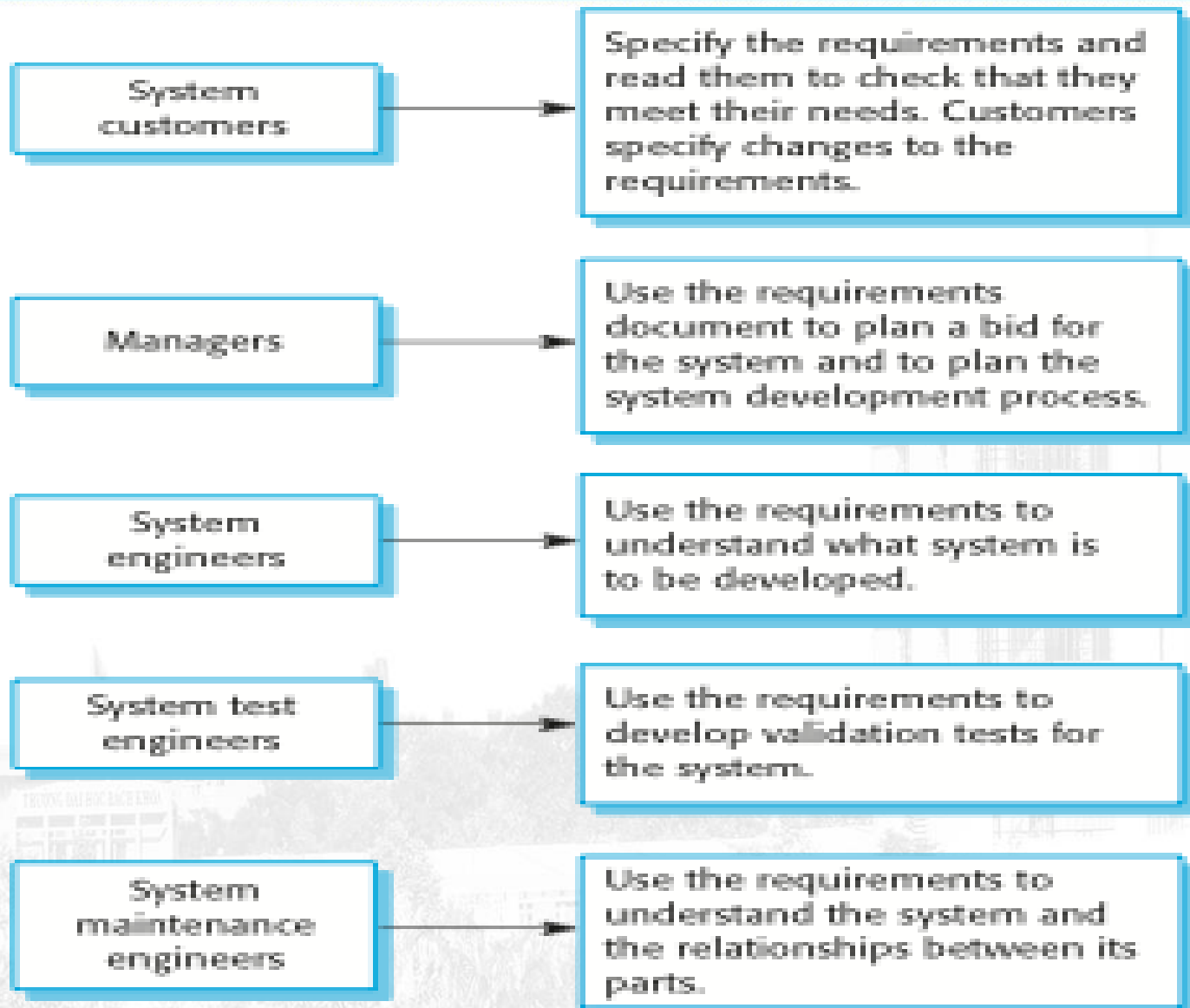
■ Khó khăn

- Khách hàng chỉ có khái niệm mơ hồ
 - Người phát triển phải chi tiết hóa thành các yêu cầu phần mềm có thể xây dựng được
- Khách hàng rất hay thay đổi yêu cầu
 - Người phát triển cần tỉnh táo để xác định đúng các yêu cầu
- Các yêu cầu thường mang tính đặc thù
 - Khó hiểu, khó định nghĩa
 - Không có chuẩn biểu diễn

■ Khó khăn

- Hệ thống đa người sử dụng
 - Các yêu cầu đa dạng, mức ưu tiên khác nhau
 - Yêu cầu mâu thuẫn nhau
- Người đặt hàng khác người sử dụng thật sự
 - Không nắm vững yêu cầu

Tài liệu yêu cầu - Người sử dụng



- Các yêu cầu của một tài liệu đặc tả yêu cầu
 - Đặc tả các hành vi bên ngoài của hệ thống
 - Đặc tả các ràng buộc cài đặt (mã hóa)
 - Dễ dàng thay đổi
 - Sử dụng như là công cụ tham khảo khi bảo trì
 - Dự báo thời gian sống của hệ thống (dự báo thay đổi)
 - Đặc tả trả lời các sự kiện không mong đợi

- Thông tin trong tài liệu yêu cầu phụ thuộc vào kiểu hệ thống và cách tiếp phát triển được sử dụng.
- Hệ thống sẽ được phát triển, sẽ có ít chi tiết hơn trong tài liệu yêu cầu.
- Yêu cầu tài liệu tiêu chuẩn đã được thiết kế, ví dụ Tiêu chuẩn IEEE. Chúng chủ yếu được áp dụng cho các yêu cầu cho các dự án kỹ thuật hệ thống lớn.

Cấu trúc tài liệu yêu cầu

Chapter	Description
Preface	This should define the expected readership of the document and describe its version history, including a rationale for the creation of a new version and a summary of the changes made in each version.
Introduction	This should describe the need for the system. It should briefly describe the system's functions and explain how it will work with other systems. It should also describe how the system fits into the overall business or strategic objectives of the organization commissioning the software.
Glossary	This should define the technical terms used in the document. You should not make assumptions about the experience or expertise of the reader.
User requirements definition	Here, you describe the services provided for the user. The nonfunctional system requirements should also be described in this section. This description may use natural language, diagrams, or other notations that are understandable to customers. Product and process standards that must be followed should be specified.
System architecture	This chapter should present a high-level overview of the anticipated system architecture, showing the distribution of functions across system modules. Architectural components that are reused should be highlighted.

Cấu trúc tài liệu yêu cầu

Chapter	Description
System requirements specification	This should describe the functional and nonfunctional requirements in more detail. If necessary, further detail may also be added to the nonfunctional requirements. Interfaces to other systems may be defined.
System models	This might include graphical system models showing the relationships between the system components and the system and its environment. Examples of possible models are object models, data-flow models, or semantic data models.
System evolution	This should describe the fundamental assumptions on which the system is based, and any anticipated changes due to hardware evolution, changing user needs, and so on. This section is useful for system designers as it may help them avoid design decisions that would constrain likely future changes to the system.
Appendices	These should provide detailed, specific information that is related to the application being developed; for example, hardware and database descriptions. Hardware requirements define the minimal and optimal configurations for the system. Database requirements define the logical organization of the data used by the system and the relationships between data.
Index	Several indexes to the document may be included. As well as a normal alphabetic index, there may be an index of diagrams, an index of functions, and so on.

- Quá trình viết ra các yêu cầu của người dùng và hệ thống trong một tài liệu yêu cầu.
- Yêu cầu người dùng phải có thể hiểu được bởi người dùng cuối và khách hàng không có nền tảng kỹ thuật.
- Yêu cầu hệ thống là yêu cầu chi tiết hơn và có thể bao gồm nhiều thông tin kỹ thuật hơn.
- Các yêu cầu có thể là một phần của hợp đồng phát triển hệ thống
 - Do đó, điều quan trọng là những điều này là đầy đủ nhất có thể.

Cách viết đặc tả yêu cầu hệ thống

Notation	Description
Natural language	The requirements are written using numbered sentences in natural language. Each sentence should express one requirement.
Structured natural language	The requirements are written in natural language on a standard form or template. Each field provides information about an aspect of the requirement.
Design description languages	This approach uses a language like a programming language, but with more abstract features to specify the requirements by defining an operational model of the system. This approach is now rarely used although it can be useful for interface specifications.
Graphical notations	Graphical models, supplemented by text annotations, are used to define the functional requirements for the system; UML use case and sequence diagrams are commonly used.
Mathematical specifications	These notations are based on mathematical concepts such as finite-state machines or sets. Although these unambiguous specifications can reduce the ambiguity in a requirements document, most customers don't understand a formal specification. They cannot check that it represents what they want and are reluctant to accept it as a system contract

- Yêu cầu được viết dưới dạng phát biểu ngôn ngữ tự nhiên được bổ sung bằng sơ đồ và bảng.
- Được sử dụng để viết các yêu cầu bởi vì nó là biểu cảm, trực quan và phổ quát. Điều này có nghĩa là các yêu cầu có thể được hiểu bởi người dùng và khách hàng.

- Đề xuất một định dạng tiêu chuẩn và sử dụng nó cho tất cả các yêu cầu.
- Sử dụng ngôn ngữ một cách nhất quán. Sử dụng phải cho các yêu cầu bắt buộc, nên cho các yêu cầu mong muốn.
- Sử dụng từ vựng văn bản để xác định các phần chính của yêu cầu.
- Tránh sử dụng thuật ngữ máy tính.
- Bao gồm một lời giải thích (lý do) tại sao một yêu cầu là cần thiết.

- Về nguyên tắc, các yêu cầu phải nêu rõ hệ thống nên làm gì và thiết kế nên mô tả cách thức thực hiện việc này.
- Trong thực tế, các yêu cầu và thiết kế không thể tách rời
 - Một kiến trúc hệ thống có thể được thiết kế để cấu trúc các yêu cầu;
 - Hệ thống có thể tương tác với các hệ thống khác mà sinh ra yêu cầu thiết kế;
 - Việc sử dụng một kiến trúc cụ thể để đáp ứng các yêu cầu phi chức năng có thể là một yêu cầu miền.
 - Đây có thể là kết quả của một yêu cầu quy định.

Cấu trúc của tài liệu yêu cầu



- Giới thiệu
- Thuật ngữ
- Định nghĩa yêu cầu người sử dụng
- Kiến trúc hệ thống
- Đặc tả yêu cầu hệ thống
- Mô hình hệ thống
- Phát triển/thay đổi của hệ thống
- Phụ lục
- Chỉ mục

Định dạng của tài liệu yêu cầu



1. Introduction

- 1.1 Purpose
- 1.2 Document Conventions
- 1.3 Intended Audience and Reading Suggestions
- 1.4 Product Scope
- 1.5 References

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

3. External Interface Requirements

- 3.1 User Interfaces
- 3.2 Hardware Interfaces
- 3.3 Software Interfaces
- 3.4 Communications Interfaces

4. System Features

- 4.1 System Feature 1
- 4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements

- 5.1 Performance Requirements
- 5.2 Safety Requirements
- 5.3 Security Requirements
- 5.4 Software Quality Attributes
- 5.5 Business Rules

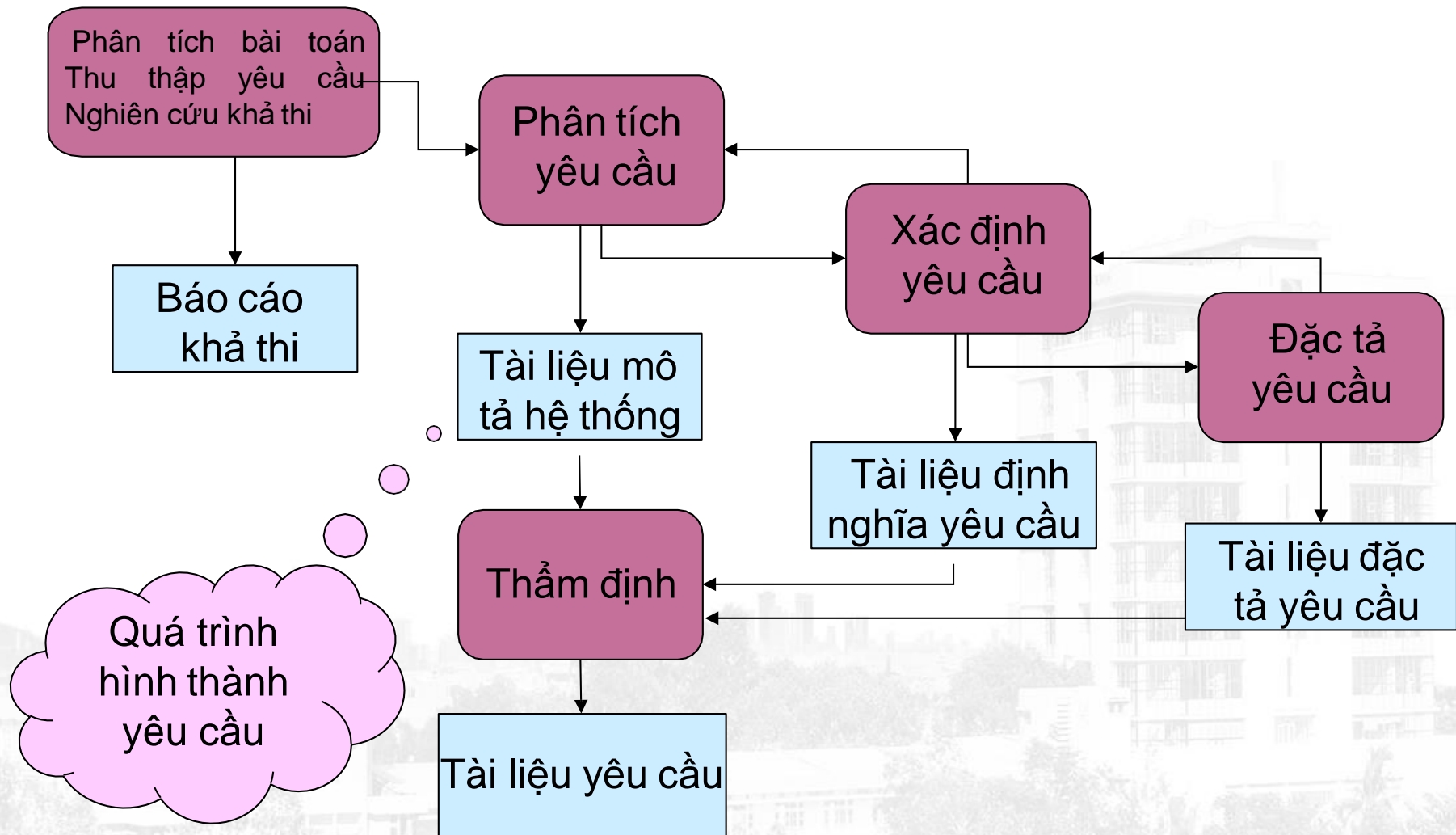
6. Other requirement

Appendix A: Glossary

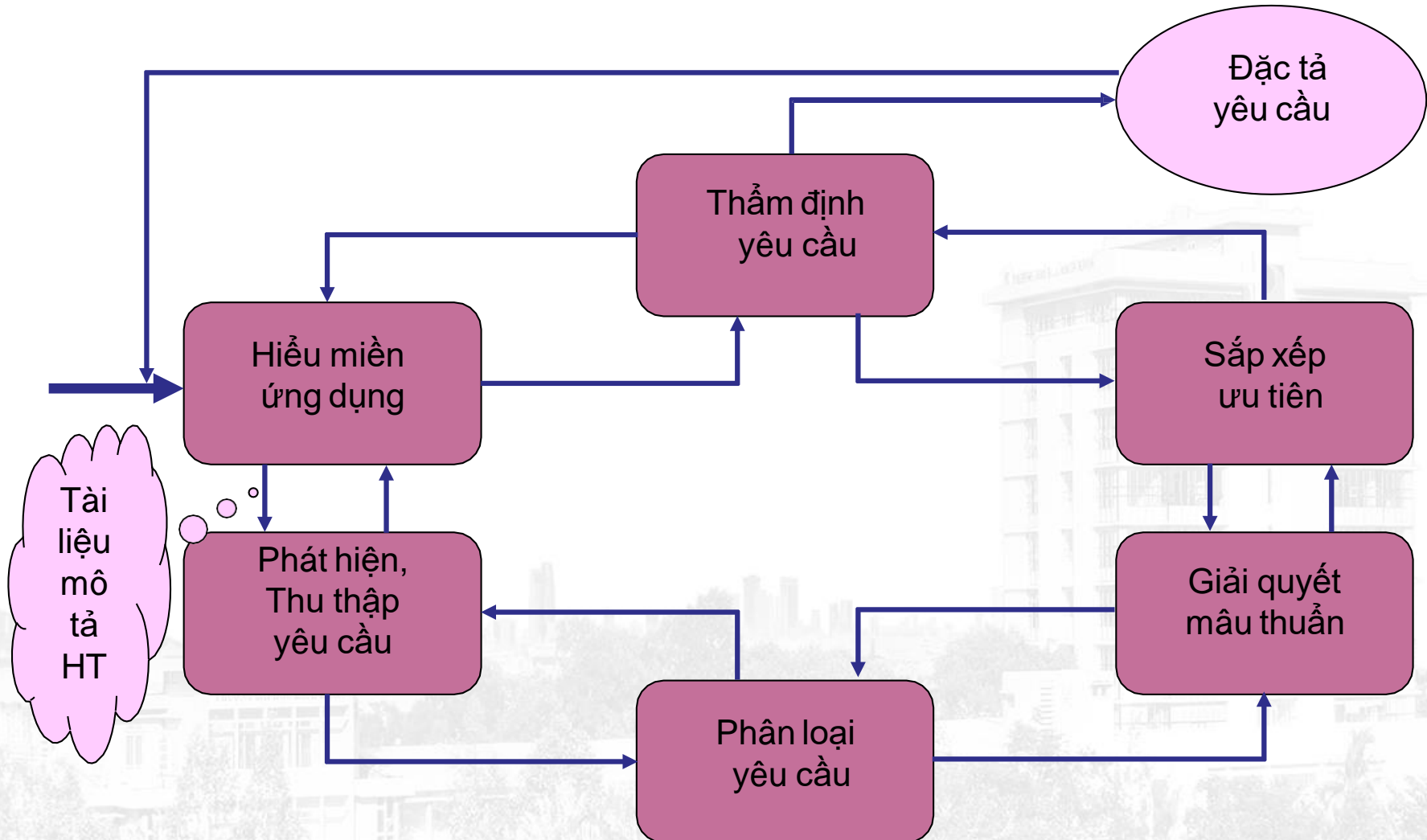
Appendix B: Analysis Models

Appendix C: To Be Determined List

Các công đoạn phân tích và đặc tả



Tiến trình phân tích yêu cầu



■ Mô tả nghiệp vụ

- Mô tả các luồng nghiệp vụ, các xử lý và vai trò của con người trong hệ thống hiện tại
- Hiểu được nghiệp vụ
- Chủ yếu tập trung vào các miền cần tự động hoá
- Hỗ trợ cho việc xác định các thay đổi và cải tiến yêu cầu trong hệ thống mới

■ Mô tả hệ thống

- Mô tả hệ thống đề xuất
 - Mô tả luồng thông tin giữa hệ thống đề xuất và môi trường của nó
- Đáp ứng được các mô tả nghiệp vụ
- Cải tiến nghiệp vụ hiện tại
- Dựa trên mô tả nghiệp vụ hiện tại

- Xác định tính khả thi của hệ thống
 - Kinh tế
 - Kỹ thuật
 - Vận hành
- Xác định những người liên quan đến hệ thống và những người sử dụng cuối
- Xác định các ràng buộc khi sử dụng hệ thống đề xuất

- Xác định phương pháp thu thập yêu cầu
 - Phỏng vấn
- Xác định các yêu cầu nhập nhằng
 - Có thể sử dụng kỹ thuật nguyên mẫu
- Xác định các yêu cầu khác mà khách hàng không yêu cầu rõ
 - Ví dụ: giao diện dễ sử dụng

- Kết quả của bước thu thập yêu cầu
 - Phát biểu về sự cần thiết và tính khả thi
 - Giới hạn lĩnh vực/chức năng của phần mềm
 - Danh sách những người liên quan/sử dụng phần mềm
 - Mô tả môi trường sẽ vận hành phần mềm
 - Danh sách các yêu cầu phần mềm đề xuất
 - Các ràng buộc phần mềm đề xuất

- Các kỹ thuật thu thập yêu cầu
- Phỏng vấn
- Thực hiện các cuộc họp/hội thảo
- Chuẩn bị bảng câu hỏi điều tra
 - Chức năng mong đợi
 - Thời gian yêu cầu hoàn thành dự án
 - Kết quả một tiến trình nghiệp vụ
- Quan sát hoạt động nghiệp vụ hiện tại
 - Đến nơi làm việc của khách hàng để khảo sát, quay phim,...
- Tham khảo các chuyên gia trong lĩnh vực
 - Hiểu rõ các nghiệp vụ chuyên môn phức tạp

■ Phỏng vấn khách hàng (1)

- Hiểu rõ nghiệp vụ hiện tại
- Hiểu rõ chi tiết yêu cầu
- Hiểu rõ mong muốn thực sự của khách hàng
- Chuẩn bị
 - Lập danh sách đối tượng, hẹn gặp, lập kế hoạch, chuẩn bị câu hỏi, phương tiện,...

■ Câu hỏi

- Câu hỏi đóng – mở
- Câu hỏi ngắn gọn, tập trung vào chủ đề

■ Cách thức tiến hành

- Theo nhóm, ít nhất hai người
- Phân công hỏi và ghi chép

■ Phỏng vấn khách hàng (2)

- Câu hỏi
 - Câu hỏi đóng – mở
 - Câu hỏi ngắn gọn, tập trung vào chủ đề
 - * Ví dụ
 - Những ai sử dụng hệ thống ?
 - Kết quả của chức năng này là gì?

■ Cách thức tiến hành

- Theo nhóm, ít nhất hai người
- Phân công hỏi và ghi chép

■ Phỏng vấn khách hàng (2)

- Các hoạt động cần thiết cho phỏng vấn
 - Xác định rõ những người cần phỏng vấn
 - Chuẩn bị sẵn các câu hỏi
 - Tìm hiểu về lĩnh vực hoạt động của hệ thống, của khách hàng
 - Ghi nhận các câu hỏi trong quá trình phỏng vấn

- Thực hiện các hội thảo/thảo luận
 - Tập hợp khách hàng, những người liên quan đến hệ thống
 - Tổ chức các buổi thảo luận
 - Trình bày các yêu cầu của hệ thống cần phát triển
 - Khách hàng có hiểu yêu cầu ?
 - Khuyến khích ý kiến của khách hàng

- Chuẩn bị các bảng câu hỏi điều tra
 - Chuẩn bị sẵn bảng các câu hỏi
 - Chức năng mong đợi
 - Thời gian yêu cầu hoàn thành dự án
 - Kết quả của một tiến trình nghiệp vụ
 - Hỏi được nhiều người
- Quan sát hoạt động nghiệp vụ hiện tại đến nơi làm việc của khách hàng và quan sát quay phim các nghiệp vụ
- Tham khảo các chuyên gia trong lĩnh vực hiểu rõ các nghiệp vụ chuyên môn phức tạp

- Mục tiêu của nghiên cứu khả thi là đi đến kết luận:
 - Có nên phát triển hệ thống hay không?
- Nội dung nghiên cứu khả thi tập trung vào những câu hỏi sau:
 - Hệ thống xây dựng sẽ giúp gì cho tổ chức?
 - Hệ thống xây dựng sử dụng công nghệ nào và kinh phí bao nhiêu?
 - Hệ thống cần tích hợp với hệ thống nào đang sử dụng?

- Báo cáo khả thi được viết dựa trên thông tin, báo cáo thu thập được, những đánh giá ban đầu về hệ thống hiện tại và phác họa các phương án dự kiến.
- Câu hỏi đặt ra cho người của tổ chức:
 - Cái gì xảy ra nếu hệ thống không được triển khai?
 - Những vấn đề gì đang đặt ra cần giải quyết?
 - Hệ thống được đề xuất giúp họ như thế nào?
 - Những tích hợp gì cần phải có?
 - Công nghệ mới gì, kỹ năng gì cần có?
 - Những tiện ích gì cần sự trợ giúp từ hệ thống?

- Đôi khi còn gọi là **phát hiện yêu cầu**.
- Nhà kỹ thuật cùng với khách hàng (người dùng, kỹ sư, nhà quản lý, chuyên gia,...-người liên quan) làm rõ **lĩnh vực ứng dụng, các dịch vụ mà hệ thống cần cung cấp và các ràng buộc trong hoạt động của nó**.
- Xây dựng các mô hình phân tích để làm rõ các yêu cầu miền.

■ Những khó khăn của phân tích

- Khách hàng thường mơ hồ về các yêu cầu, không biết mình muốn cụ thể gì, dễ lẫn lộn giữa yêu cầu và mong muốn.
- Họ thể hiện yêu cầu theo thuật ngữ riêng
- Khách hàng đa dạng có thể đưa ra các yêu cầu mâu thuẫn.
- Những yếu tố tổ chức và chính sách có thể ảnh hưởng đến yêu cầu
- Yêu cầu thường mang tính đặc thù, khó hiểu, khó có chuẩn chung.
- Các yêu cầu thay đổi trong quá trình phân tích: môi trường nghiệp vụ thay đổi,...

- Phân loại yêu cầu
 - Chức năng
 - Phi chức năng
- Yêu cầu chức năng xuất phát từ các yêu cầu của khách hàng và nghiệp vụ trong hệ thống hiện tại
- Yêu cầu phi chức năng thường không lộ rõ
 - Thường do người phát triển đề xuất

- Mục tiêu, mong muốn và yêu cầu
- Mục tiêu, mong muốn: là cái hướng tới.
 - Ví dụ: “Xây dựng giao diện thân thiện”
- Yêu cầu: là cái cụ thể kiểm tra được
 - Ví dụ: “Giao diện đồ họa có các lệnh được chọn bằng menu”
- Nhiệm vụ của người phân tích là các xác định đúng, đầy đủ và chính xác các yêu cầu.

Nguyên lý 1: Mô hình hóa miền thông tin

- Phải hiểu và biểu diễn được miền thông tin (problem domain)
 - Định danh dữ liệu (đối tượng, thực thể)
 - Định nghĩa các thuộc tính
 - Thiết lập các mối quan hệ giữa các dữ liệu
- => Từ điển dữ liệu, mô hình thực thể mối quan hệ, mô hình khái niệm

- Nguyên lý 2: **Mô hình hóa chức năng**
- Bản chất của phần mềm là biến đổi thông tin
 - Định danh các chức năng (*biến đổi thông tin*)
 - Xác định cách thức dữ liệu (*thông tin*) di chuyển trong hệ thống (*luồng dữ liệu*)
 - Xác định các tác nhân tạo dữ liệu (*nguồn*) và tác nhân tiêu thụ dữ liệu (*đích*)

=> Mô hình phân rã chức năng, mô hình luồng dữ liệu

Nguyên lý 3: Mô hình hóa hành vi

- Phần mềm (hệ thống) có trạng thái (hành vi)
 - Xác định các trạng thái của hệ thống
 - Ví dụ: giao diện đồ họa, section trong ứng dụng web
 - Xác định các dữ kiện làm thay đổi hành vi hệ thống
 - Ví dụ: bàn phím, chuột, các cổng thông tin...

=> Biểu đồ trạng thái

Nguyên lý 4: Phân hoạch các mô hình

- Làm mịn, phân hoạch và biểu diễn các mô hình ở các mức khác nhau
 - Làm mịn các mô hình dữ liệu
 - Tạo cây (mô hình) phân rã chức năng
 - Biểu diễn hành vi ở các mức chi tiết khác nhau
- => Mô hình luồng dữ liệu các mức 1,2,3,..

Nguyên lý 5: Tìm hiểu vấn đề bản chất

- Nhìn nhận bản chất của yêu cầu (*làm gì, điều kiện gì,...*)
- Không quan tâm đến cách thức cài đặt (*làm như thế nào*)

Có thể phân hoạch yêu cầu theo hai cách

■ Phân loại theo đặc trưng:

- Yêu cầu tương hỗ: chịu ảnh hưởng của môi trường
- Yêu cầu nảy sinh: nhận ra trong quá trình phát triển
- Yêu cầu hệ quả: là kết quả của việc áp dụng hệ thống dựa trên máy tính
- Yêu cầu tương thích: phụ thuộc vào hệ khác hay tiến trình tổ chức

■ Phân loại theo mức độ quan trọng: chính, phụ

- Mô tả chi tiết các yêu cầu đã phân tích
- Có thể sử dụng các cấu trúc tài liệu đặc tả yêu cầu khác nhau
 - Chẳng hạn cấu trúc IEEE
- Tuy nhiên, phải chứa ít nhất các thông tin
 - Định nghĩa hệ thống phần mềm
 - Mục đích tài liệu đặc tả yêu cầu
 - Giới hạn của hệ thống phần mềm
 - Yêu cầu chức năng
 - Yêu cầu phi chức năng
- Các điều kiện mà trong đó hệ thống đề xuất sẽ vận hành

- Thẩm định yêu cầu liên quan tới việc kiểm tra tính đúng đắn, tính đầy đủ, tính hiện thực và kiểm tra được của yêu cầu.
 1. Thỏa mãn được nhu cầu của người dùng?
 2. Yêu cầu không mâu thuẫn nhau?
 3. Yêu cầu phải đầy đủ (chức năng, ràng buộc)?
 4. Yêu cầu là hiện thực?
 5. Yêu cầu có thể kiểm tra được?

- Chỉ ra rằng các yêu cầu thực sự là cái khách hàng cần
- Lỗi ở bước đặc tả yêu cầu chi phí rất lớn
 - Chi phí sửa một lỗi yêu cầu sau khi đã giao sản phẩm có thể lớn gấp 100 lần lỗi cài đặt
- Kỹ thuật nguyên mẫu rất hiệu quả để hợp thức hóa yêu cầu

- Thường xuyên thẩm định yêu cầu
- Cả khách hàng và người phát triển đều phải thẩm định yêu cầu
- Việc thẩm định có thể tổ chức hình thức hoặc không hình thức
- Trao đổi giữa khách hàng, nhà phát triển và người sử dụng cuối có thể giải quyết sớm các khó khăn

Kiểm tra các tính chất

- Hợp lệ
 - Hệ thống phần mềm có cung cấp các chức năng hỗ trợ tốt nhất cho khách hàng?
- Chắc chắn
 - Có các yêu cầu nào mâu thuẫn nhau?
- Đầy đủ
 - Tất cả các yêu cầu của khách hàng đã được đặc tả?
- Thực tế
 - Tất cả các yêu cầu có thể thực hiện với công nghệ và ngân sách hiện tại?

Các kỹ thuật thẩm định

1. Xem xét lại yêu cầu: bằng cách phân tích một cách có hệ thống các yêu cầu (có sự kết hợp cả nhà phát triển và khách hàng, tiến hành thường xuyên)
2. Làm bản mẫu: sử dụng các mô hình có khả năng thực thi để kiểm tra lại yêu cầu
3. Tạo các trường hợp kiểm thử: kiểm tra tính có thể kiểm tra được của yêu cầu
4. Phân tích tính nhất quán bằng cách tự động: sử dụng CASE (Computer Aided Software Engineering)

Đặc tả yêu cầu phần mềm



- Đặc tả các yêu cầu phần mềm là công việc xây dựng các tài liệu đặc tả
 - Có thể sử dụng tới các công cụ như: mô hình hóa, mô hình toán học hình thức (a formal mathematical model), tập hợp các kịch bản sử dụng, các nguyên mẫu hoặc bất kỳ một tổ hợp các công cụ nói trên
- Đặc tả chi tiết các yêu cầu đã phân tích
- Chất lượng của hồ sơ đặc tả đánh giá qua các tiêu thức
 - Tính rõ ràng, chính xác
 - Tính phù hợp
 - Tính đầy đủ, hoàn thiện

- Có thể sử dụng các cấu trúc tài liệu đặc tả yêu cầu khác nhau (IEEE)
- Đầy đủ
 - Mọi yêu cầu của người dùng phải được mô tả
- Không mâu thuẫn nhau
- Chính xác : yêu cầu không được mơ hồ
 - Chi phí phát sinh cho sửa đổi rất cao

Các mức trừu tượng của yêu cầu



- Yêu cầu nên được biểu diễn ở nhiều mức trừu tượng khác nhau phục vụ cho nhiều đối tượng khác nhau
- Xác định yêu cầu
 - Mô tả các dịch vụ mà phần mềm cần cung cấp
 - Viết bằng ngôn ngữ tự nhiên
 - Hướng người dùng
- Đặc tả yêu cầu
 - Tài liệu có cấu trúc
 - Mô tả chi tiết, chính xác về yêu cầu
 - Dùng làm bản hợp đồng

Ví dụ: Chức năng kiểm tra lỗi chính tả



=> Định nghĩa:

- Thông báo các lỗi chính tả của văn bản

=> Đặc tả:

- Các lỗi chính tả được gạch đỏ bên dưới
- Các lỗi soạn thảo được gạch xanh bên dưới
- Lỗi chính tả:
 - Từ đơn không có trong từ điển
- Lỗi soạn thảo
 - Thừa dấu cách
 - Không viết hoa đầu câu

Các thành phần của hồ sơ đặc tả



- Đặc tả phi hình thức (Informal specifications) được viết bằng ngôn ngữ tự nhiên
- Đặc tả hình thức (Formal specifications) được viết bằng tập các ký pháp có các quy định về cú pháp (syntax) và ý nghĩa (semantic) rất chặt chẽ
- Đặc tả vận hành chức năng (Operational specifications) mô tả các hoạt động của hệ thống phần mềm sẽ xây dựng
- Đặc tả mô tả (Descriptive specifications) – đặc tả các đặc tính đặc trưng của phần mềm

