



## Nội dung chính

1

Yêu cầu phần mềm

2

Thu thập yêu cầu

3

Phân tích yêu cầu

4

Đặc tả yêu cầu



## Yêu cầu phần mềm

- Yêu cầu (requirement) là những mô tả các thuộc tính **đầy đủ và cần thiết** để một sản phẩm hay dịch vụ **thoả mãn** yêu cầu khách hàng.
- Xác định yêu cầu là quá trình **xác định các dịch vụ** khách hàng yêu cầu từ hệ thống và những **ràng buộc** cho hệ thống hoạt động, phát triển.



## Phân loại yêu cầu

- **Yêu cầu chức năng** (functional requirement): mô tả những gì (**WHAT**) hệ thống làm được như đăng nhập, tra cứu, báo cáo, thống kê, v.v.
- **Yêu cầu phi chức năng** (non-functional requirement) là những yêu cầu không liên quan đến dịch vụ được chỉ định, mà liên quan đến những thuộc tính **ràng buộc** hệ thống như tính tin cậy, tính an hiệu quả, tính tiện dụng, v.v.



## Khảo sát hiện trạng

- Khảo sát hiện trạng nhằm tìm hiểu **thế giới thực** liên quan đến phần mềm.
- Các hiện trạng cần khảo sát
  - Hiện trạng tổ chức
  - Hiện trạng tin học
  - Hiện trạng nghiệp vụ
- Các kỹ thuật khảo sát hiện trạng: phỏng vấn, khảo sát bằng bảng câu hỏi, nghiên cứu tài liệu, quan sát thực tế.



## Thu thập yêu cầu

- Thu thập yêu cầu làm việc với **stakeholder** nhằm xác định **phạm vi hệ thống**, các **ng nghiệp vụ** hệ thống cần cung cấp và những **ràng buộc** hoạt động hệ thống.
  - Stakeholder là một người hoặc nhóm người có liên quan đến phần mềm, ảnh hưởng yêu cầu phần mềm hoặc có ảnh hưởng đến sản phẩm phần mềm.



## Thu thập yêu cầu

- Các phương pháp thu thập yêu cầu:
  - Phỏng vấn
    - Phỏng vấn cá nhân/phỏng vấn nhóm.
    - Phỏng vấn tự do/phỏng vấn có định hướng.
  - Bảng câu hỏi
  - Nghiên cứu tài liệu
  - Quan sát thực tế



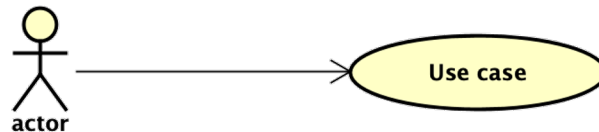
## Phân tích yêu cầu

- Phân tích yêu cầu nhằm xác định những **đặc trưng phù hợp cho sản phẩm** dựa trên nhu cầu của stakeholder.
- Stakeholder và developer thường có khung nhìn và cách mô tả khác nhau, nên để phát tích hiệu quả kỹ sư phần mềm cần dùng những mô hình trừu tượng dễ dàng hiểu giữa các bên.
- **Lược đồ use case** thường được sử dụng trong phân tích yêu cầu.



## Lược đồ Use case

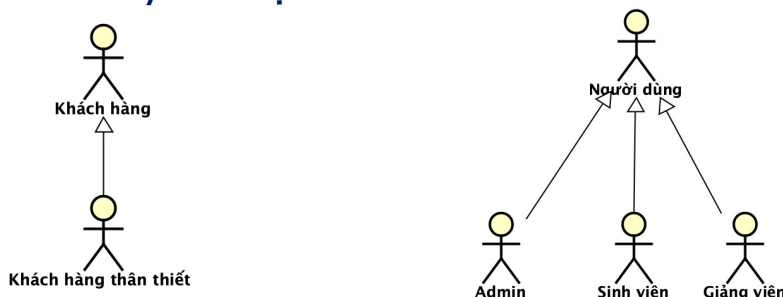
- Lược đồ use case nhằm mô tả các chức năng gì của hệ thống được thực hiện bởi actor nào và vai trò các actor trong hệ thống.
- Lược đồ use case thể hiện sự tương tác giữa actor và use case.



- Chiều mũi tên thể hiện sự chủ động tương tác.

## Actor

- Actor là tác nhân bên ngoài hệ thống và có tương tác với hệ thống.
- Actor có thể là con người, phần cứng hoặc một phần mềm khác.
- Giữa hai actor có thể có mối quan hệ tổng quát hoá và chuyên biệt hoá.





## Làm thế nào xác định actor?

- Ai hoặc tổ chức sử dụng các chức năng của hệ thống?
- Ai hoặc tổ chức là người cung cấp, sử dụng hoặc lấy thông tin từ hệ thống?
- Ai là người duy trì, bảo dưỡng và quản lý hệ thống?
- Tài nguyên bên ngoài hệ thống là gì?
- Có hệ thống nào khác tương tác với hệ thống này?

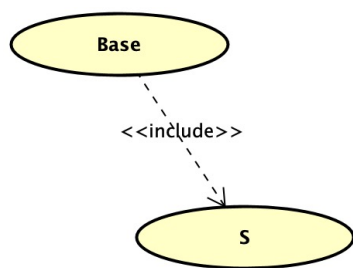


## Use case

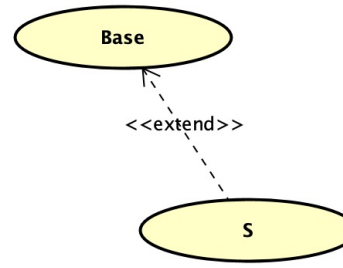
- Use case là một chuỗi các hành động mà hệ thống có thể thực hiện, tương tác với actor hệ thống.
- Mỗi use case được xem là một chức năng hệ thống mang lại một kết quả nhất định cho người dùng.

## Use case

- Giữa hai Use case có thể có quan hệ <<include>> hoặc <<extend>>



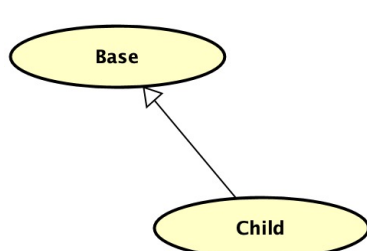
Use Case S luôn được thực hiện khi Use case Base được gọi



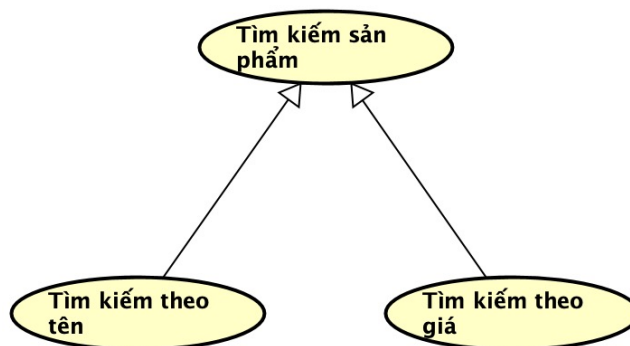
Use Case S thỉnh thoảng được thực hiện khi Use case Base được gọi

## Use case

- Quan hệ tổng quát hoá, chuyên biệt hoá



Use case Child là trường hợp chuyên biệt, nâng cấp của use case Base

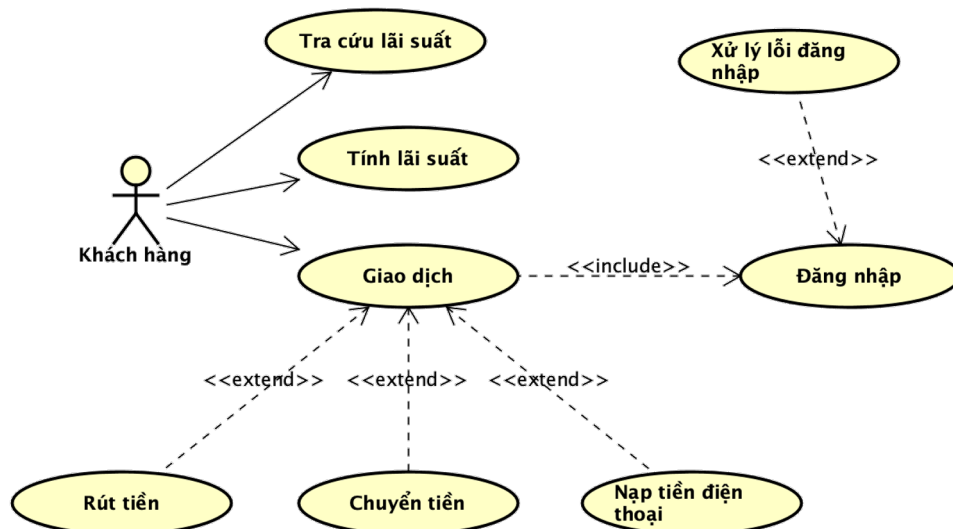


## Làm thế nào xác định use case?

- Với mỗi actor, xác định những việc liên quan đến actor?
- Hệ thống hỗ trợ những nghiệp vụ nào trong thế giới thực?
- Những thông tin nào cần được quản lý trong hệ thống?
- Những thông tin nào cần được kết xuất khỏi hệ thống?

## Lược đồ Use case

- Ví dụ







## Đặc tả Use case

- Use case Id
- Tên use case
- Mô tả vắn tắt use case
- Actor chính (primary actor): actor chính tương tác để thực hiện các chức năng hệ thống và nhận được kết quả nhất định từ hệ thống.
- Actor phụ (secondary actor): các actor hỗ trợ để actor chính thực hiện được công việc của use case.



## Đặc tả use case

- Tiền điều kiện (pre-conditions): điều kiện ràng buộc trước khi use case được thực thi.
- Hậu điều kiện (post-conditions)
- Luồng hoạt động (main flows): những việc chính sẽ được thực hiện bởi actor.
- Luồng thay thế (alternative flows)
- Luồng ngoại lệ (exception flows)



## Đặc tả use case "Rút tiền"

- Mô tả: use case này cho phép khách hàng thực hiện rút tiền từ máy ATM.
- Actor chính: Khách hàng
- Actor phụ: Không có
- Tiền điều kiện
  - Khách hàng đã đưa thẻ vào máy và đã chứng thực mật khẩu.
- Hậu điều kiện
  - Gửi tin nhắn thông báo khách hàng về số dư trong tài khoản đã có thay đổi.



## Đặc tả use case "Rút tiền"

- Luồng hoạt động
  1. Khách hàng chọn giao dịch "Rút tiền".
  2. Hệ thống yêu cầu nhập mật khẩu lần nữa.
  3. Hệ thống yêu cầu chọn hoặc nhập số tiền muốn rút.
  4. Hệ thống kiểm tra đủ tiền không.
  5. Khách hàng có thể tiếp tục chọn giao dịch khác hoặc chọn "Huỷ giao dịch" để kết thúc.



## Đặc tả use case “Rút tiền”

- Luồng thay thế
  - Ở bước 2 người dùng được nhập mật khẩu tối đa 3 lần, nếu 3 lần đều sai hệ thống quay về màn hình chính.
  - Ở bước 4 nếu đủ tiền hệ thống sẽ nhả tiền và quay về màn hình chính. Ngược lại, hệ thống báo lỗi và quay về màn hình chính.
- Luồng ngoại lệ: không có
  - Nếu hệ thống có sự cố trong quá trình thực hiện thì máy nhả thẻ cho KH.



## Đặc tả yêu cầu

- Đặc tả yêu cầu là quá trình sàng lọc, tinh chế, tổ chức các yêu cầu thành tài liệu.
- “A specification can be a written **document**, a **graphical model**, a **formal mathematical model**, a **collection of usage scenarios**, a **prototype**, or a **combination of these**.”  
Roger Pressman
- “A requirements specification describes the **functions** and **performance of a computer** based system and the **constraints** that will govern its development.”

Alan Davis



## Đặc tả yêu cầu

- Các loại tài liệu đặc tả cơ bản
  - Business Requirements Document (BRD)
  - User's Requirements Document (URD)
  - System Requirements Document (SRD)
  - Functional Requirements Document (FRD)
  - Architecture Design Document (ADD)
  - Hardware Requirements Documentt (HRD)
  - **Software Requirements Specification (SRS)**
  - Interfaces Definition Document (IDD)



## Đặc tả yêu cầu

- Tài liệu đặc tả yêu cầu (Software Requirement Specification – SRS) là các yêu cầu mà nhà phát triển phần mềm thiết kế, lập trình, kiểm thử các giải pháp phần mềm.
- SRS chứa các yêu cầu chức năng và phi chức năng mà sản phẩm phần mềm cần đáp ứng.
- SRS được xem như hợp đồng giữa khách hàng và tổ chức phát triển phần mềm.



## Đặc tả yêu cầu

- Các phương pháp đặc tả yêu cầu
  - Dùng **ngôn ngữ hình thức** (formal specification) dùng giống như ngôn ngữ lập trình hoặc các công thức toán học. Phương pháp này rõ ràng, chính xác. Tuy nhiên cách tiếp cận này phức tạp và hạn chế người đọc và hiểu.
  - Dùng **mô hình** dùng các hình vẽ mô tả yêu cầu. Phương pháp này ngắn gọn, súc tích.



## Đặc tả yêu cầu

- Tài liệu SRS phải bao gồm
  - Chuyển từ các mô hình phân tích và các thông tin yêu cầu từ người dùng **thành các thành phần yêu cầu phần mềm**.
  - **Kết hợp các mô hình** phân tích trực tiếp vào đặc tả yêu cầu phần mềm
  - Cung cấp những **chi tiết cần thiết để thiết kế và viết mã nguồn** cho sản phẩm.
  - Cung cấp thông tin cần bản **viết test plan và các thủ tục**.
  - Một cơ sở để tạo ra các sổ tay và tài liệu **hướng dẫn người dùng**.
  - Chỉ định **độ ưu tiên** của các yêu cầu.



## Đặc tả yêu cầu

- Cấu trúc tài liệu đặc tả yêu cầu
  - Lời tựa (perface)
  - Giới thiệu (introduction)
  - Thuật ngữ (glossary)
  - Định nghĩa các yêu cầu người dùng
  - Kiến trúc hệ thống
  - Đặc tả yêu cầu hệ thống
  - Các mô hình hệ thống
  - Sự tiến hoá hệ thống
  - Các phụ lục (appendices)
  - Chỉ mục (index)

