

Unified Modeling Language

Khảo sát yêu cầu với
mô hình Use Case

Giảng viên: Mai Thúy Nga



Nội dung môn học

Giới thiệu về PTTK hướng đối tượng với UML 1	Khái niệm về Hướng đối tượng trong UML 2	Đặc tả Yêu cầu với mô hình Use Case I 3
Đặc tả Yêu cầu với mô hình Use Case II 4	Tổng quan về Phân tích và Thiết kế 5	Phân tích Use Case I 6
Phân tích Use Case II 7	Mô hình hóa Thiết kế 8	Ôn tập 9

Requirement Engineering using Use Case

3/4 - 3

Nội dung trình bày

- ☆ Mục đích của phân tích yêu cầu
 - ♦ Khái niệm về mô hình Use Case
 - ♦ Xây dựng mô hình Use Case trong quá trình phân tích
 - ♦ Đặc tả Use Case

Requirement Engineering using Use Case

3/4 - 4

Mục đích của phân tích yêu cầu (1)

- ♦ Thiết lập và duy trì sự thoả thuận giữa khách hàng và người tham gia dự án về việc hệ thống sẽ làm được những gì
 - Không nói làm như thế nào để đạt được điều đó
- ♦ Giúp cho những người phát triển hệ thống một sự hiểu biết rõ hơn về những yêu cầu của hệ thống
- ♦ Đưa ra những giới hạn mà hệ thống sẽ thực hiện và KHÔNG thực hiện
- ♦ Cung cấp các thông tin cơ bản để lập kế hoạch phát triển dự án

Requirement Engineering using Use Case

3/4 - 5

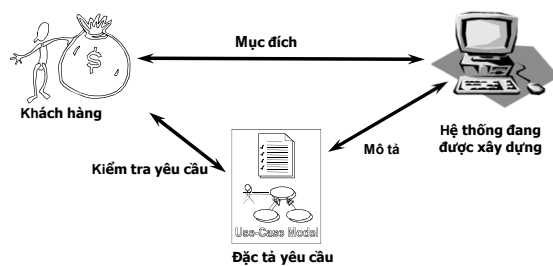
Mục đích của phân tích yêu cầu (2)

- ♦ Cung cấp những cơ sở để ước lượng giá thành và thời gian để phát triển hệ thống
- ♦ Xác định được giao diện người dùng của hệ thống, nắm bắt được những yêu cầu và mục đích của người sử dụng
- ♦ Là đầu vào cho giai đoạn tiếp theo của dự án: Phân tích và thiết kế

Requirement Engineering using Use Case

3/4 - 6

Mục đích của phân tích yêu cầu (3)



Requirement Engineering using Use Case

3/4 - 7

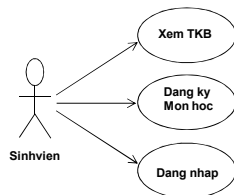
Nội dung trình bày

- ♦ Mục đích của phân tích yêu cầu
- ☆ Khái niệm về mô hình Use Case
- ♦ Xây dựng mô hình Use Case trong quá trình phân tích
- ♦ Đặc tả Use Case

Requirement Engineering using Use Case 3/4 - 8

Mô hình Use-Case (1)

- ♦ Mô hình hóa các chức năng mà hệ thống sẽ thực thi
 - Mô hình bao gồm các chức năng định trước của hệ thống
 - Sử dụng khái niệm Use Case



Requirement Engineering using Use Case 3/4 - 9

Mô hình Use-Case (2)

- ♦ Đóng vai trò như một thỏa thuận giữa khách hàng, người sử dụng và những người phát triển hệ thống
 - Cho phép khách hàng và người sử dụng kiểm tra những chức năng mà họ mong đợi hệ thống sẽ thực hiện
 - Người phát triển có thể hiểu được cần phải làm gì
- ♦ Mỗi Use-Case được mô tả chi tiết, chỉ ra từng bước tác nhân và hệ thống tương tác với nhau, hệ thống làm gì cho tác nhân trong Use Case đó
- ♦ Cần xác định rõ phạm vi của hệ thống
 - Các chức năng hệ thống sẽ đảm nhiệm
 - Các chức năng được thực hiện bên ngoài hệ thống

Requirement Engineering using Use Case 3/4 - 10

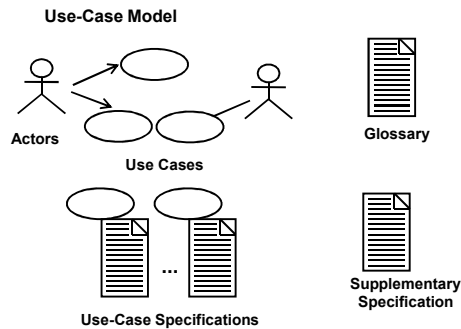
Các đối tượng sử dụng mô hình Use Case

- ♦ Người sử dụng sử dụng sơ đồ use-case để hiểu thêm về hệ thống
- ♦ Người thiết kế sử dụng sơ đồ use-case làm cơ sở cho những công việc của họ và nắm bắt tổng quan hệ thống
- ♦ Người kiểm tra chương trình sử dụng nó để lập kế hoạch cho các tác vụ kiểm tra
- ♦ Người viết tài liệu sử dụng nó làm cơ sở để viết hướng dẫn cho người sử dụng
- ♦ Quản lý dự án sử dụng nó để lập kế hoạch
- ♦ ...

Requirement Engineering using Use Case

3/4 - 11

Các thành phần chính trong mô hình Use Case



Requirement Engineering using Use Case

3/4 - 12

Tác nhân và Use-Case

- ♦ Tác nhân (Actor)
 - Tác nhân thể hiện "một cái gì đó" tương tác với hệ thống
- ♦ Chức năng (Use-Case)
 - Mô tả chức năng mà hệ thống có
 - Mục đích là để PHÂN TÍCH yêu cầu nghiệp vụ của bài toán chứ không phải để THIẾT KẾ phần mềm

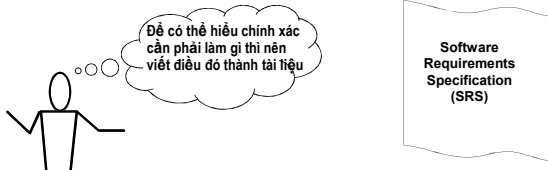


Requirement Engineering using Use Case

3/4 - 13

Bản đặc tả Use-Case (Use-Case Specification)

- ♦ Bản đặc tả Use-Case
 - Là một tài liệu bao gồm tất cả các đặc tính của use-case đã được văn bản hoá
 - Ví dụ: Miêu tả tóm tắt, luồng sự kiện...



Để có thể hiểu chính xác cần phải làm gì thì nên viết điều đó thành tài liệu

Software Requirements Specification (SRS)

Requirement Engineering using Use Case 3/4 - 14

Thuật ngữ và Đặc tả hỗ trợ

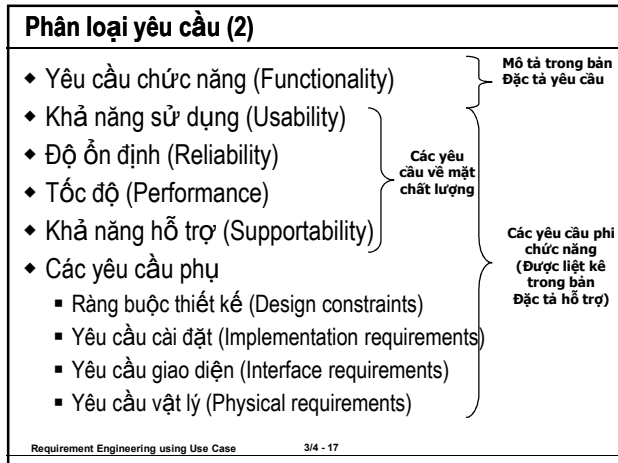
- ♦ Các thuật ngữ (Glossary):
 - Định nghĩa, giải thích rõ những từ chuyên môn, thuật ngữ chung sử dụng trong các tài liệu
- ♦ Đặc tả hỗ trợ (Supplementary Specification):
 - Đặc tả các yêu cầu không liên quan trực tiếp trong Use-Case (những yêu cầu không chính quy), nhưng cũng rất quan trọng để đảm bảo chất lượng sản phẩm:
 - Độ tin cậy
 - Chất lượng...

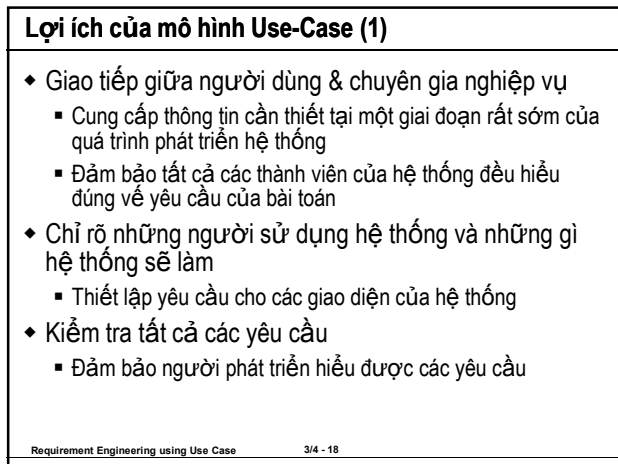
Requirement Engineering using Use Case 3/4 - 15

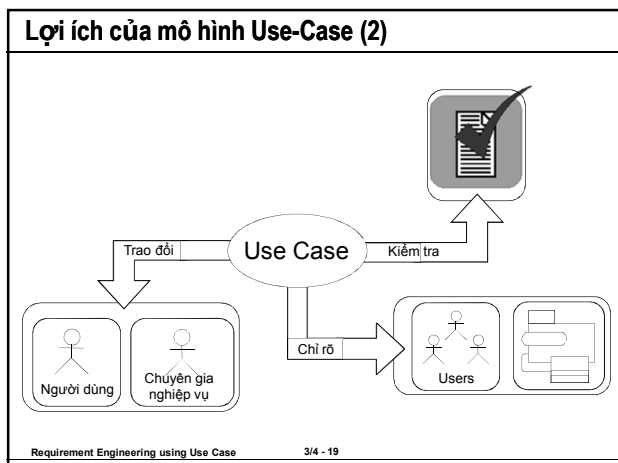
Phân loại yêu cầu (1)

- ♦ Yêu cầu chức năng
 - Mô tả các hành động mà hệ thống có thể làm được mà không quan tâm đến các ràng buộc vật lý khác (ví dụ không nói PC tốc độ bao nhiêu, RAM bao nhiêu...)
 - Các yêu cầu này được mô tả trong bản Đặc tả Use Case
 - Cần chỉ rõ đầu vào/ra của chức năng này trong hệ thống
- ♦ Yêu cầu phi chức năng
 - Chỉ mô tả các thuộc tính của hệ thống cũng như môi trường của hệ thống (ví dụ nói rõ cấu hình, trình duyệt, phần mềm...)
 - Được mô tả trong phần Đặc tả hỗ trợ

Requirement Engineering using Use Case 3/4 - 16





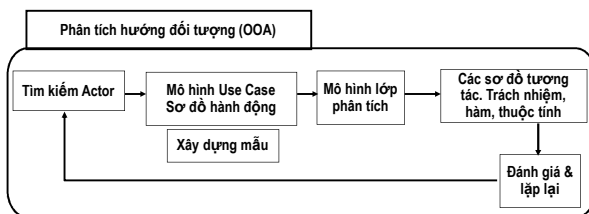


Nội dung trình bày

- ♦ Mục đích của phân tích yêu cầu
- ♦ Khái niệm về mô hình Use Case
- ☆ Xây dựng mô hình Use Case trong quá trình phân tích
- ♦ Đặc tả Use Case

Requirement Engineering using Use Case 3/4 - 20

Phân tích hướng đối tượng (OOA)



Requirement Engineering using Use Case 3/4 - 21

Quy trình trong OOA

- ♦ Tìm kiếm tác nhân (Actors)
 - Ai sẽ sử dụng hệ thống?
- ♦ Phát triển Use Cases
 - Người dùng sẽ làm gì với hệ thống?
- ♦ Phát triển các mô hình quy trình nghiệp vụ sử dụng UML
 - Sơ đồ hành động (Activity Diagrams)
- ♦ Xây dựng sơ đồ lớp phân tích sử dụng UML
 - Tìm kiếm các thực thể
- ♦ Phát triển sơ đồ tương tác (interaction diagram)
 - Sơ đồ trình tự công việc cần thiết để hoàn thành công việc cụ thể
 - Gán cho từng lớp các kiểm quan hệ, thuộc tính, hàm
- ♦ Đánh giá và lặp lại
 - Trong trường hợp cần thiết, lặp lại các bước trên để hoàn thiện

Requirement Engineering using Use Case 3/4 - 22

Tìm tác nhân của hệ thống (1)

- ♦ Tác nhân (Actor):
 - Các thực thể bên ngoài mà hệ thống phải tương tác với
 - Thực thể này có thể là một cá nhân, cũng có thể là một hệ thống khác mà chúng ta không phải xây dựng
 - Ví dụ như các thiết bị ngoại vi, thậm chí là database
- ♦ Thông tin về tác nhân:
 - Tên tác nhân phải mô tả vai trò của tác nhân đó một cách rõ ràng
 - Tên nên là danh từ
 - Cần mô tả khái quát khả năng của tác nhân đó

Requirement Engineering using Use Case

3/4 - 23

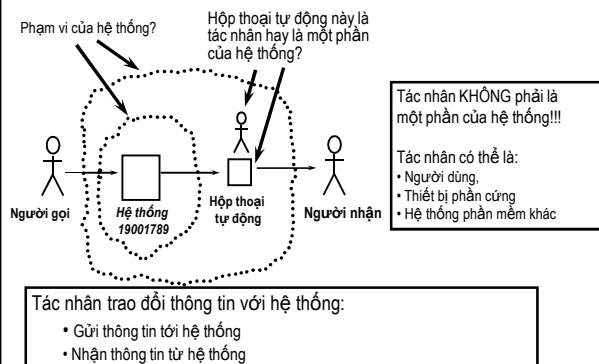
Tìm tác nhân của hệ thống (2)

- ♦ Đặt các câu hỏi sau để tìm ra tác nhân:
 - Nhóm người nào yêu cầu hệ thống làm việc giúp họ?
 - Nhóm người nào kích hoạt chức năng của hệ thống?
 - Nhóm người nào sẽ duy trì và quản trị hệ thống hoạt động?
 - Hệ thống có tương tác với các thiết bị hay phần mềm ngoại vi nào khác hay không?
 - Ngoại vi: chúng ta không xây dựng chúng

Requirement Engineering using Use Case

3/4 - 24

Ví dụ về tác nhân



Requirement Engineering using Use Case

3/4 - 25

Tìm chức năng của hệ thống

- ♦ Chức năng (Use Case):
 - Các tác nhân yêu cầu những gì từ hệ thống?
 - Xem các yêu cầu chức năng để tìm ra các Use Case
- ♦ Đối với mỗi tác nhân tìm được, đặt các câu hỏi:
 - Các công việc chính mà tác nhân đó muốn HT thực thi?
 - Tác nhân đó có tạo ra hay thay đổi dữ liệu gì của HT?
 - Tác nhân đó có phải thông báo gì cho HT?
 - Tác nhân đó có cần thông tin thông báo gì từ HT?
- ♦ Thông tin về Use Case:
 - Tên của UC nên chỉ rõ kết quả của quá trình tương tác với tác nhân
 - Tên nên là động từ
 - Mô tả ngắn gọn về mục đích của UC

Requirement Engineering using Use Case

3/4 - 26

Những điều nên tránh khi làm Use Case

- ♦ Tạo ra các Use Case quá nhỏ
 - Hành động quá đơn giản mà chỉ cần mô tả bởi vài dòng
- ♦ Tạo ra quá nhiều Use case (hàng chục)
 - Nhóm các Use case liên quan thành một Use case tổng quát (mức 1)
 - Mô tả các Use Case tổng quát ở một sơ đồ khác (mức 2)
 - Ví dụ: "Quản lý sách" bao gồm "Nhập sách", "Xuất sách", "..."
- ♦ Sử dụng các Use-case quá cụ thể, hoặc làm việc với dữ liệu quá cụ thể. Ví dụ:
 - "Tìm sách theo tên" (nên là "Tìm sách")
 - "Nhập Pin vào máy ATM" (nên là "Đăng nhập")
 - "Thêm sách" (nên là "Quản lý sách" bao gồm "Thêm sách")

Requirement Engineering using Use Case

3/4 - 27

Tương tác giữa Tác nhân và Use Case

- ♦ Thiết lập quan hệ giữa Tác nhân và Use Case
 - Chúng tương tác bằng cách gửi các tín hiệu cho nhau
- ♦ Chiều của quan hệ chính là chiều của tín hiệu gửi đi
 - Từ tác nhân tới Use Case
 - Kích hoạt Use case
 - Hỏi thông tin nào đó trong hệ thống
 - Thay đổi thông tin nào đó trong hệ thống
 - Thông báo cho Use Case về một sự kiện đặc biệt nào đó xảy ra với hệ thống
 - Từ Use Case tới Actor
 - Nếu như có một điều gì đó xảy ra với HT và tác nhân đó cần được biết sự kiện đó
 - Use Case đôi khi cần hỏi thông tin nào đó từ một tác nhân trước khi Use Case đó đưa ra một quyết định



Requirement Engineering using Use Case

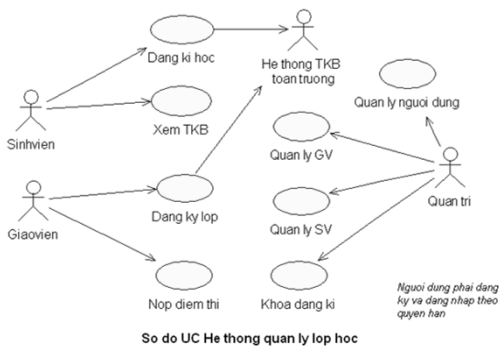
3/4 - 28

Nội dung trình bày

- ♦ Mục đích của phân tích yêu cầu
- ♦ Các nhân tố liên quan đến yêu cầu
- ♦ Xây dựng mô hình Use Case trong quá trình phân tích
- ☆ Đặc tả Use Case

Requirement Engineering using Use Case 3/4 - 29

Thử đọc một sơ đồ (1)



Requirement Engineering using Use Case 3/4 - 30

Thử đọc một sơ đồ (2)

- ♦ Trả lời các câu hỏi sau:
 - Mô tả các chức năng của hệ thống
 - Sinh viên có thể tác động lên những use-case nào?
 - Giáo viên có thể tác động lên những use-case nào?
 - Nếu A vừa là sinh viên vừa là giáo viên, anh ta có thể thực hiện được những use-case nào?
 - Sơ đồ này không nói lên được những gì?
 - Những use-case nào cần thiết thực hiện đầu tiên?
- ♦ Sơ đồ Use Case có thể mô tả được hết không?

Requirement Engineering using Use Case 3/4 - 31

Đặc tả Use-Case
<ul style="list-style-type: none"> ♦ Tên: <ul style="list-style-type: none"> ▪ Tên của Use-case ♦ Mô tả ngắn gọn: <ul style="list-style-type: none"> ▪ Mô tả về vai trò và mục đích của use-case, tránh kiểu diễn xuôi tên Use Case ♦ Điều kiện <ul style="list-style-type: none"> ▪ Tiền điều kiện ▪ Hậu điều kiện ♦ Luồng sự kiện (kịch bản): <ul style="list-style-type: none"> ▪ Mô tả bằng lời những gì mà hệ thống sẽ làm thể hiện trên use-case ♦ Sơ đồ hoạt động <ul style="list-style-type: none"> ▪ Minh họa luồng sự kiện bằng mô hình ♦ Các yêu cầu đặc biệt ♦ ...
<small>Requirement Engineering using Use Case 3/4 - 32</small>

Luồng sự kiện của use-case (1)
<ul style="list-style-type: none"> ♦ Trả lời được quá trình từ khi bắt đầu đến khi kết thúc của một use-case <ul style="list-style-type: none"> ▪ Chỉ mô tả chi tiết các sự kiện thuộc use-case đó ▪ Nếu có sự liên hệ với Use Case khác, nên có sự phân tích và tham khảo ngắn gọn ♦ Mô tả dữ liệu được trao đổi giữa tác nhân và use-case đó <div style="border: 1px solid black; padding: 5px; margin: 5px;"> <ul style="list-style-type: none"> ▪ Cấu trúc: Ai làm gì, khi nào, với dữ liệu gì, [vì mục đích gì] ▪ Cần phân tích rõ hệ thống cần phải làm gì để đáp ứng được yêu cầu của tác nhân đó. Không được mặc định cho rằng hệ thống tự biết làm điều đó </div> <ul style="list-style-type: none"> ▪ Tránh mô tả chức năng, g/diện hoặc chi tiết từng thành phần dữ liệu ▪ Tránh mô tả chung chung, hoặc lúc nào cũng đúng
<small>Requirement Engineering using Use Case 3/4 - 33</small>

Luồng sự kiện của use-case (2)
<ul style="list-style-type: none"> ♦ Luồng chính (Basic flow) <ul style="list-style-type: none"> ▪ Luồng lý tưởng mà Use case thường hoạt động ♦ Luồng phát sinh (Alternative flow) <ul style="list-style-type: none"> ▪ Sử dụng nhiều lần trong luồng chính ▪ Các trường hợp đặc biệt (vd nhấn mạnh một tính năng của HT) ▪ Gây ra lỗi, cách xử lý lỗi trong tình huống đó ♦ Chú ý <ul style="list-style-type: none"> ▪ Chỉ cần luồng chính là có thể hiểu được tác vụ chính mà Use Case đó sẽ thực thi ▪ Phải có lời gọi luồng phát sinh từ luồng chính ▪ Tránh viết luồng phát sinh quá dài, hoặc dài hơn luồng chính ▪ Tránh tách quá nhiều luồng phát sinh
<small>Requirement Engineering using Use Case 3/4 - 34</small>

Luồng sự kiện của use-case (3)

♦ Kịch bản là một thể hiện của Use Case đó

- Một Use Case có nhiều kịch bản tùy thuộc vào ngữ cảnh cụ thể mà nó phát sinh

Requirement Engineering using Use Case 3/4 - 35

Sơ đồ hành động (Activity Diagram)

♦ Sơ đồ hành động được sử dụng để minh họa luồng sự kiện

Requirement Engineering using Use Case 3/4 - 36

Một số ký hiệu khác trong mô hình Use Case

♦ Các quan hệ

- Giao tiếp
- Bao gồm (include)
- Mở rộng (extend)
- Tổng quát hoá (generalization)

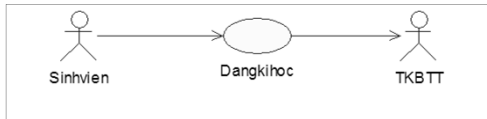
♦ Ghi chú (Notes)

♦ Tạo các gói (package)

Requirement Engineering using Use Case 3/4 - 37

Quan hệ giao tiếp (1)

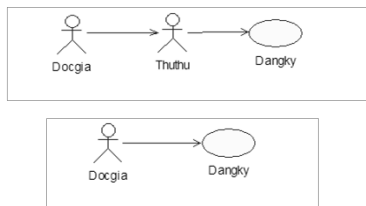
- ♦ Là quan hệ giữa UC và tác nhân, thể hiện bằng mũi tên



Requirement Engineering using Use Case 3/4 - 38

Quan hệ giao tiếp (2)

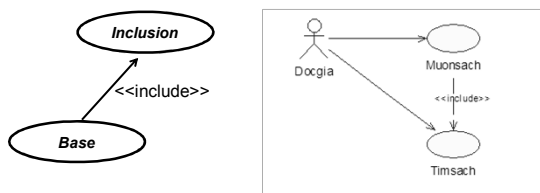
- ♦ Xét sự khác nhau trong 2 sơ đồ sau:



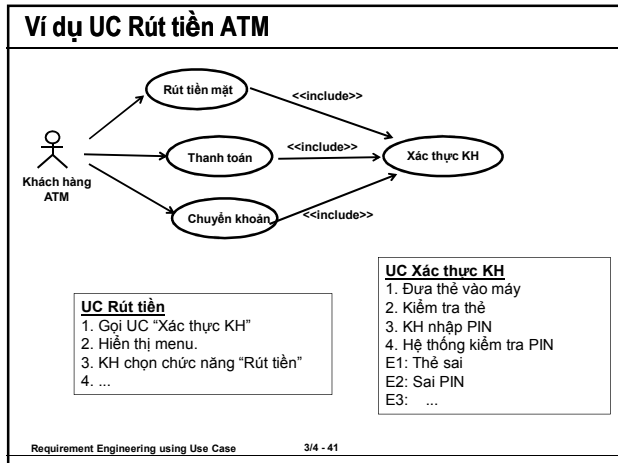
Requirement Engineering using Use Case 3/4 - 39

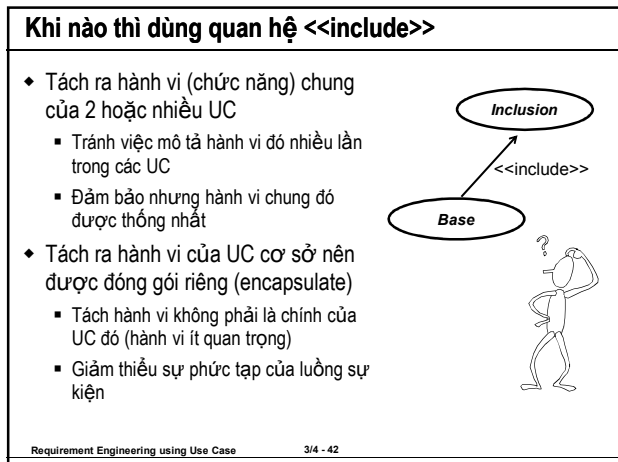
Quan hệ bao gồm (include)

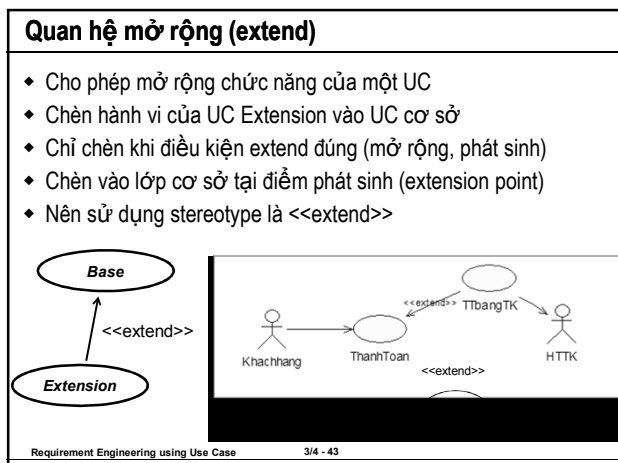
- ♦ Cho phép một UC sử dụng chức năng của UC khác
- ♦ Chức năng của UC Inclusion sẽ được gọi trong UC cơ bản
- ♦ Nên sử dụng stereotype là <<include>>

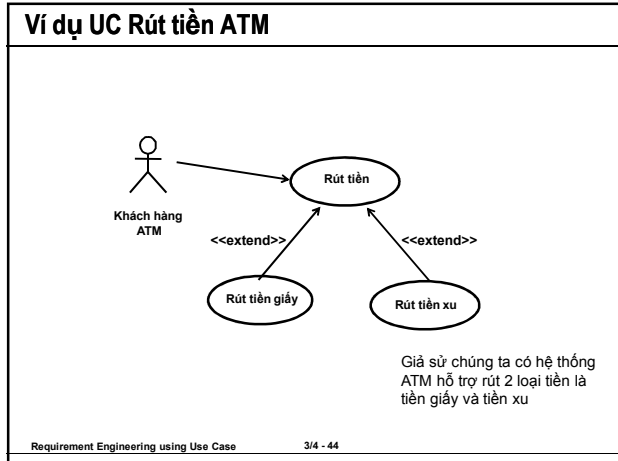


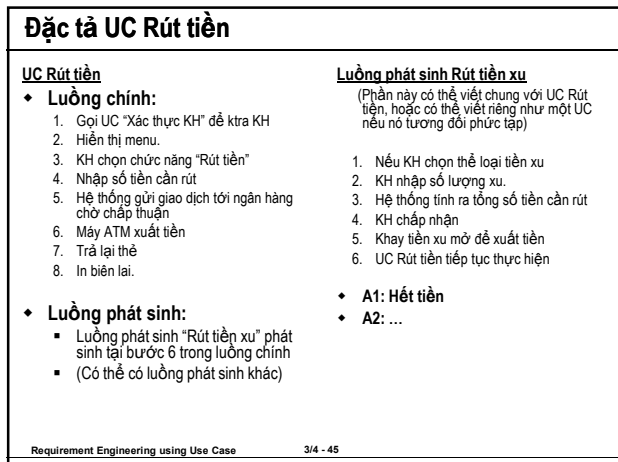
Requirement Engineering using Use Case 3/4 - 40

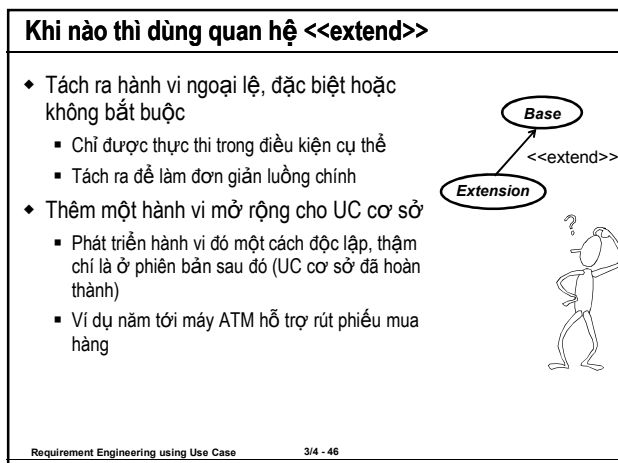






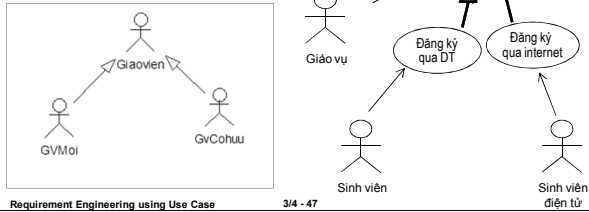






Quan hệ tổng quát hoá (Generalization)

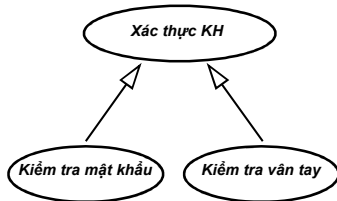
- Được sử dụng để chỉ ra một vài tính chất chung của một nhóm tác nhân hoặc Use case
- Sử dụng khái niệm kế thừa
 - Mô tả hành vi chung (chia sẻ) trong UC cha
 - Mô tả hành vi riêng trong (các) UC con



Requirement Engineering using Use Case

3/4 - 47

Ví dụ UC Rút tiền ATM

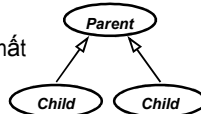


Requirement Engineering using Use Case

3/4 - 48

Khi nào thì dùng quan hệ <<generalization>>

- Dùng để mô tả các hành vi chung, cấu trúc và mục đích trong 2 hay nhiều UC
 - Chỉ ra UC con là một thành phần của họ UC
 - Tránh mô tả hành vi (chung) nhiều lần
 - Đảm bảo hành vi chung được thống nhất
- Thực thi UC con
 - Theo luồng sự kiện của UC cha
 - Chèn thêm hành vi riêng của UC con theo sự mô tả trong UC con
 - Một số hành vi trong UC cha có thể bị sửa đổi trong UC con

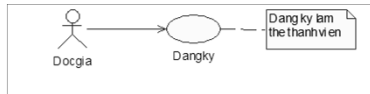


Requirement Engineering using Use Case

3/4 - 49

Thêm ghi chú (Notes)

- ♦ Thêm các ghi chú để sơ đồ rõ ràng hơn



Requirement Engineering using Use Case

3/4 - 50

Tạo các gói (Package)

- ♦ Có thể nhóm các thành phần (Use-Case, Actor, quan hệ hoặc các sơ đồ khác) thành một nhóm chung (package)
- ♦ Nếu số lượng Use Case quá lớn, nên chia chúng vào các nhóm
 - Dễ hiểu mô hình tổng thể hơn
 - Dễ bảo trì mô hình Use Case
 - Dễ giao việc cho các thành viên
- ♦ Xem xét khả năng gộp nhóm
 - Tương tác với cùng một tác nhân
 - Nhóm Use Case hợp thành một quy trình (module) tương đối hoàn thiện

Requirement Engineering using Use Case

3/4 - 51

Tóm tắt

- ♦ Đặc tả yêu cầu
 - Đóng vai trò như một thỏa thuận giữa khách hàng, người sử dụng và những người phát triển hệ thống
 - Cho phép khách hàng và người sử dụng kiểm tra những chức năng mà họ mong đợi hệ thống sẽ thực hiện
 - Người phát triển có thể hiểu được cần phải làm gì
 - Cần tuân thủ theo một quy trình hợp lý
- ♦ Một số chú ý khi đặc tả yêu cầu với mô hình Use Case
 - Sử dụng mẫu tài liệu, hướng dẫn
 - Phát triển các luồng sự kiện đơn giản, rõ ràng về mặt nghiệp vụ (tránh đưa ra các thông tin quá chi tiết, kỹ thuật)
 - Xây dựng sơ đồ hành động cho những vấn đề phức tạp

Requirement Engineering using Use Case

3/4 - 52

Ví dụ về UC Mua hàng trên mạng	
♦ Mô tả:	<ul style="list-style-type: none"> Giả sử có một hệ thống của hàng áo trên mạng UC Bán hàng cho phép khách hàng (KH) mua được các mặt hàng mong muốn Ví dụ này yêu cầu KH phải thanh toán trực tuyến
♦ Tiền điều kiện:	<ul style="list-style-type: none"> KH muốn mua hàng trên cửa hàng ảo KH có thể thanh toán điện tử tới ngân hàng mà cửa hàng hỗ trợ
♦ Hậu điều kiện:	<ul style="list-style-type: none"> Thành công khi KH chấp nhận mua hàng và quá trình thanh toán với ngân hàng thực hiện thành công. Hóa đơn được lập, hàng hóa được dành riêng cho KH đó Nếu quá trình thanh toán với ngân hàng không thành công, hóa đơn sẽ không được lập, hàng cũng không được bán ra
♦ Thực thể:	<ul style="list-style-type: none"> Mặt hàng, Giỏ hàng, Đơn hàng
♦ Use case liên quan:	<ul style="list-style-type: none"> Tìm kiếm hàng, quản lý đơn hàng (Giao hàng)
Requirement Engineering using Use Case 3/4 - 53	

Luồng sự kiện cho Use Case	
1.	KH duyệt, tìm kiếm và xem thông tin các mặt hàng muốn mua (xem Use-Case xem hàng)
1.	KH có thể chọn chức năng "Đưa hàng vào giỏ hàng"
2.	Hệ thống sẽ đưa mặt hàng này vào giỏ
3.	KH có thể nhập số lượng muốn mua (mặc định là 1)
4.	Hệ thống sẽ tự động cập nhật giá của giỏ hàng hiện tại
2.	KH có thể lặp lại quá trình này để mua tiếp các mặt hàng khác
1.	Giỏ hàng sẽ không mất đi trong quá trình KH tìm/mua mặt hàng khác
2.	Nếu giỏ hàng đã có mặt hàng này, hệ thống sẽ báo lại cho KH...
3.	Quản lý giỏ hàng
1.	Mỗi một KH có một giỏ hàng riêng rẽ và không ai nhìn thấy thông tin của nhau
2.	KH có thể chọn chức năng "Xem giỏ hàng" bất kỳ lúc nào cần
3.	Hệ thống sẽ hiển thị giỏ hàng với đầy đủ các mặt hàng KH đã chọn, cùng số lượng và giá cả từng loại
4.	KH có thể thay đổi số lượng, hoặc bỏ đi mặt hàng mà KH không muốn mua
4.	KH có thể chọn chức năng thanh toán, xem luồng phụ "Thanh toán"
Requirement Engineering using Use Case 3/4 - 54	

Luồng phụ: Thanh toán	
1.	KH có thể chọn chức năng thanh toán
2.	KH được yêu cầu nhập thẻ thanh toán và địa chỉ giao hàng (???)
3.	Thông tin thanh toán được đưa tới ngân hàng, hệ thống sẽ chờ kết quả từ ngân hàng đó
1.	(Quá trình xử lý giao dịch là do ngân hàng quyết định)
4.	Nếu ngân hàng không chấp nhận giao dịch
1.	Hệ thống sẽ thông báo kết quả tới KH, yêu cầu nhập lại thông tin
5.	Nếu ngân hàng chấp nhận
1.	(Số tiền tương ứng của KH được chuyển sang tài khoản của cửa hàng)
2.	Hệ thống sẽ lập Đơn hàng và lưu lại (xem Use-Case quản lý đơn hàng)
3.	Số lượng hàng tồn kho sẽ được giảm tương ứng
4.	Hệ thống thông báo thành công cho KH trên trang web và gửi thông tin đơn hàng qua mail của KH
5.	Giỏ hàng sẽ bị xóa đi (Nếu mua tiếp, giỏ hàng sẽ được tạo mới)
Requirement Engineering using Use Case 3/4 - 55	
