

**PHÂN TÍCH
THIẾT KẾ HƯỚNG
ĐỐI TƯỢNG**

Biên soạn: ThS. Lê Trung Hiếu

MỤC LỤC

MỤC LỤC.....	ii
HƯỚNG DẪN.....	iv
BÀI 1: QUY TRÌNH RUP	1
1.1 QUY TRÌNH PHÁT TRIỂN PHẦN MỀM	1
1.2 QUY TRÌNH RUP.....	3
1.2.1 Giới thiệu	3
1.2.2 Quy trình RUP	5
1.2.3 Phát triển theo mô hình lặp	6
1.2.4 Một số workflow chính trong RUP.....	8
BÀI TẬP ĐỀ NGHỊ.....	11
BÀI 2: TỔNG QUAN UML.....	12
2.1 KHÁI NIỆM.....	12
2.2 CÁC BIỂU ĐỒ CỦA UML	14
2.2.1 Biểu đồ Usecase	14
2.2.2 Biểu đồ hoạt động – Activity diagram	17
2.2.3 Biểu đồ lớp – Class diagram	21
2.2.4 Biểu đồ tuần tự - Sequence diagram	24
2.2.5 Biểu đồ giao tiếp – Communication diagram.....	27
2.2.6 Biểu đồ trạng thái – State diagram	28
2.2.7 Biểu đồ gói – Package diagram.....	29
2.2.8 Biểu đồ thành phần – Component diagram.....	30
2.2.9 Biểu đồ triển khai – Deployment diagram	30
BÀI TẬP ĐỀ NGHỊ.....	32
BÀI 3: LẤY YÊU CẦU	34
3.1 KHÁI NIỆM.....	34
3.2 KỸ THUẬT LẤY YÊU CẦU	35

3.2.1 Phỏng vấn	36
3.2.2 Họp nhóm.....	36
3.2.3 Quan sát.....	37
3.2.4 Công việc tạm thời.....	37
3.2.5 Điều tra qua câu hỏi.....	37
3.2.6 Xem xét tài liệu	38
3.2.7 Xem xét phần mềm	38
3.3 KẾT QUẢ CỦA GIAI ĐOẠN LẤY YÊU CẦU	38
3.3.1 Kết quả	38
3.3.2 Đặc tả UC	38
BÀI TẬP ĐỀ NGHỊ.....	41
BÀI 4: PHÂN TÍCH & THIẾT KẾ	42
4.1 GIỚI THIỆU	42
4.2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG Ở TRẠNG THÁI TĨNH.....	43
4.3 PHÂN TÍCH THIẾT KẾ HỆ THỐNG Ở TRẠNG THÁI ĐỘNG	48
BÀI TẬP ĐỀ NGHỊ.....	50
TÀI LIỆU THAM KHẢO	51
PHỤ LỤC – ĐÒ ÁN THAM KHẢO	52

HƯỚNG DẪN

MÔ TẢ MÔN HỌC

Môn học Phân Tích Thiết Kế Hướng Đối Tượng Cung cấp kiến thức và kỹ năng phân tích thiết kế phần mềm theo phương pháp hướng đối tượng, cung cấp kiến thức về UML và case tool để thiết kế hệ thống.

Mục tiêu của môn học là sử dụng ngôn ngữ UML và các case tool để lấy yêu cầu, phân tích và thiết kế, viết tài liệu cho một phần mềm thực tiễn.

Sau khi kết thúc môn học, người học phải có khả năng lấy yêu cầu, phân tích, thiết kế và viết tài liệu cho một phần mềm.

NỘI DUNG MÔN HỌC

Bài 1: Giới thiệu quy trình phát triển phần mềm RUP

- Tổng quan
- Giới thiệu quy trình RUP
- Phát triển phần mềm theo quy trình lặp

Bài 2: Tổng quan về ngôn ngữ UML

- Khái niệm
- Các biểu đồ của UML

Bài 3: Lấy yêu cầu

- Khái niệm
- Kỹ thuật lấy yêu cầu
- Kết quả giai đoạn lấy yêu cầu

Bài 4: Phân tích & thiết kế

-
- Tổng quan
 - Phân tích và thiết kế ở trạng thái tĩnh
 - Phân tích và thiết kế ở trạng thái động

KIẾN THỨC TIỀN ĐỀ

Cấu trúc dữ liệu, Cơ sở dữ liệu, Lập trình trong window (Visual C++), Lập trình hướng đối tượng.

YÊU CẦU MÔN HỌC

Người học phải dự học đầy đủ các buổi lên lớp và tham gia thực hành đầy đủ. Người học phải có máy tính và sử dụng các case tool như: Astah, Star UML, Rational Rose.

CÁCH TIẾP NHẬN NỘI DUNG MÔN HỌC

Để học tốt môn này, người học cần phải tìm hiểu thông tin, kiến thức về các lĩnh vực mà phần mềm thực hiện hướng đến.

PHƯƠNG PHÁP ĐÁNH GIÁ MÔN HỌC

Môn học được đánh giá gồm:

- Điểm quá trình: 30%. Hình thức và nội dung do GV quyết định, phù hợp với quy chế đào tạo và tình hình thực tế tại nơi tổ chức học tập.
- Điểm báo cáo đồ án: 70%. Hình thức thi vẫn đáp. Sinh viên làm việc nhóm và trả lời các câu hỏi liên quan về đề tài của mình. Hình thức phân chia công việc: phân chia theo Use Case. Một nhóm tối đa 3 sinh viên.

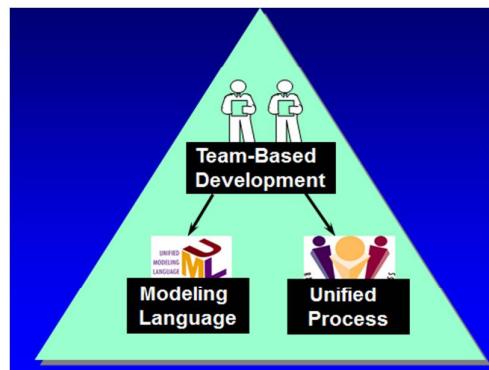
BÀI 1: QUY TRÌNH RUP

1.1 QUY TRÌNH PHÁT TRIỂN PHẦN MỀM

Quy trình phát triển phần mềm: gồm một tập hợp các hoạt động được tổ chức mà mục đích của nó là xây dựng và phát triển phần mềm:

- Quy trình xác định ai làm gì, khi nào và bằng cách nào để đạt được một mục tiêu nào đó
- Quy trình phần mềm xác định một bộ khung và tiêu chuẩn để triển khai công nghệ phần mềm

Khi xây dựng và phát triển một phần mềm, thường thì cần phải làm việc trong đội, nhóm:



Ngôn ngữ mô hình hóa hợp nhất (Unified Modeling Language - UML):

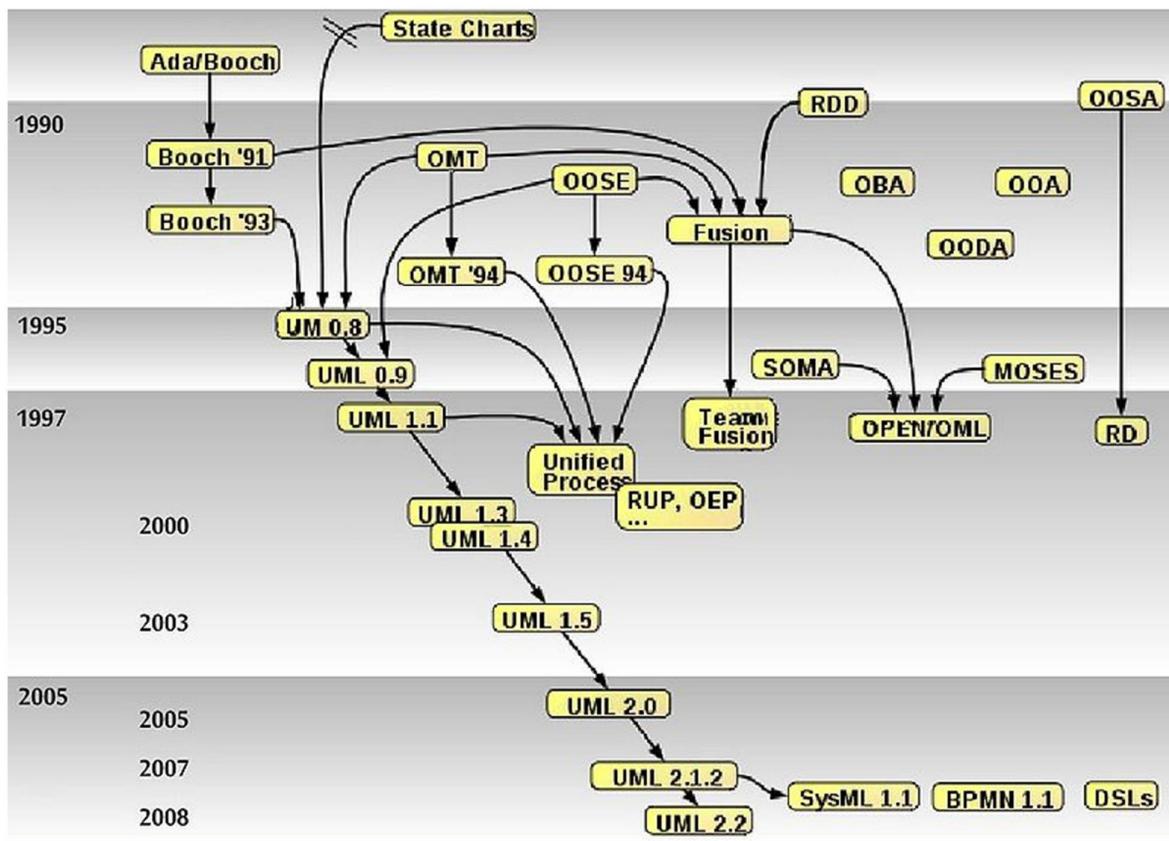
Là ngôn ngữ dùng để

- Xác định (Specifying)
- Trực quan hóa (Visualizing)
- Xây dựng (Constructing)
- Lập tài liệu (Documenting)

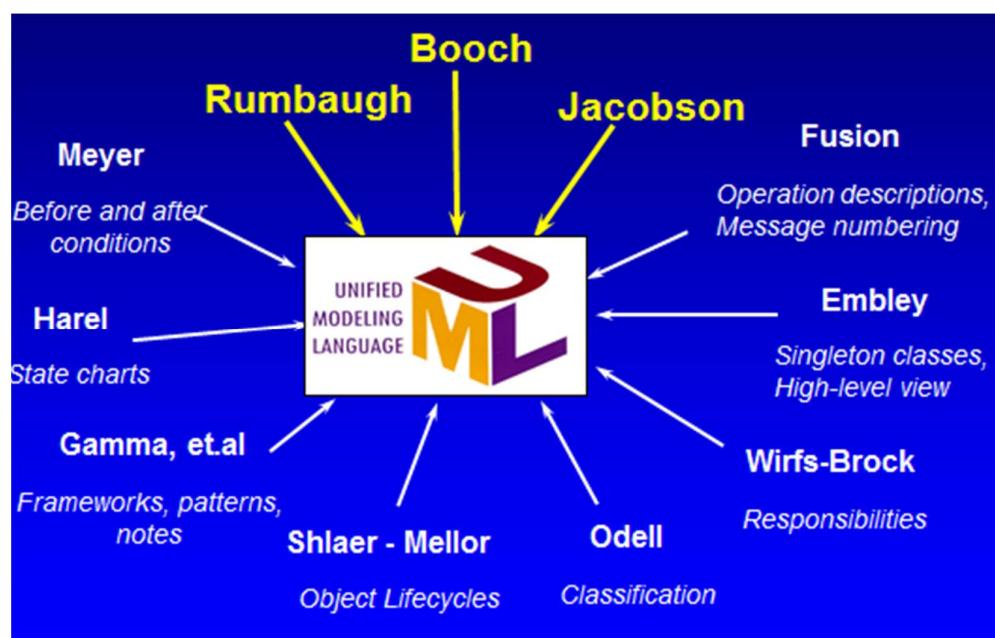


Cho các kết quả (artifacts) của quá trình thực hiện phần mềm.

Lịch sử của UML:



Những người tham gia tạo ra ngôn ngữ UML:



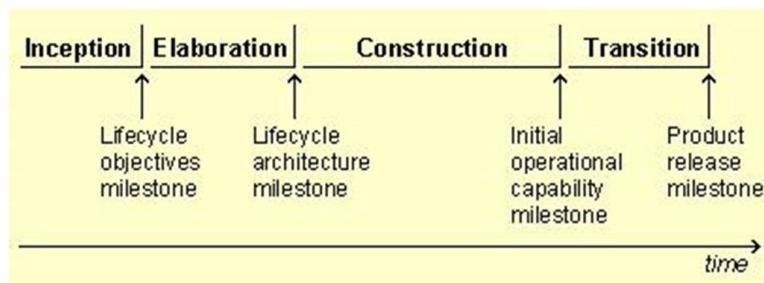


1.2 QUY TRÌNH RUP

1.2.1 GIỚI THIỆU

Quy trình phát triển phần mềm hợp nhất Rational (Rational Unified Process hay RUP) là một quy trình phát triển phần mềm. Nó cung cấp các phương pháp, các nguyên tắc phân công nhiệm vụ và trách nhiệm trong các tổ chức phát triển phần mềm. Nó là phương pháp dùng để tạo ra một sản phẩm phần mềm có chất lượng cao đảm bảo các dự thảo về thời gian và kinh phí với khách hàng.

Quy trình này được tổ chức làm bốn giai đoạn:



Từ phương diện quản lý, vòng đời của một phần mềm theo RUP được chia theo thời gian qua bốn giai đoạn nối tiếp nhau, mỗi giai đoạn có một mốc quan trọng, mỗi giai đoạn thực chất là khoảng giữa của 2 điểm mốc. Cuối mỗi giai đoạn hô nhân kiểm định sẵn thực hiện thẩm định các đổi tượng của

giai đoạn này. Nếu việc kiểm tra thích hợp thì dự án sẽ được chuyển sang giai đoạn tiếp theo.

Giai đoạn Inception: Kết quả của giai đoạn này là đạt được sự nhất trí giữa tất cả những người đóng vai trò chủ chốt về các mục tiêu của dự án. Trong giai đoạn này phải chỉ ra được các mục tiêu quan trọng cần cỗ gắng đạt được, trong đó rủi ro của các yêu cầu, các chức năng nghiệp vụ cũng phải được chỉ ra trước khi dự án bắt đầu. Trong giai đoạn này cũng đề cập đến việc cải tiến từ các hệ thống đã có, tuy nhiên vẫn chỉ tóm tắt, giai đoạn này chú trọng đến cả 2 điều quan trọng của dự án: đáng để làm hay không và khả năng thực hiện.

Mục tiêu chính của giai đoạn Inception

- Thiết lập phạm vi phần mềm và các điều kiện biên của dự án, bao gồm: nhìn nhận khả năng thực hiện, các điều kiện thỏa thuận và những gì sản phẩm mong đợi và không mong đợi
- Nhận định đúng đắn về các Use case của hệ thống, kịch bản của các hành vi trong hệ thống sẽ đóng vai trò định hướng quan trọng cho kết quả của phần thiết kế
- Trình bày, minh họa một số kiến trúc ứng cử viên cho một vài kịch bản chính
- Dự trù tất cả chi phí, lập kế hoạch cho toàn bộ dự án
- Dự trù các rủi ro tiềm ẩn
- Chuẩn bị môi trường hỗ trợ cho dự án

Giai đoạn Elaboration: Kết quả của giai đoạn này là tạo ra một baseline cho kiến trúc của hệ thống tạo cơ sở cho quá trình thiết kế và thực thi trong giai đoạn construction. Kiến trúc này mở rộng việc phân tích các yêu cầu quan trọng của hệ thống (các yêu cầu có sự ảnh hưởng lớn đến hệ thống) và

đánh giá các rủi ro. Sự ổn định của kiến trúc được đánh giá qua nhiều nguyên bản của kiến trúc.

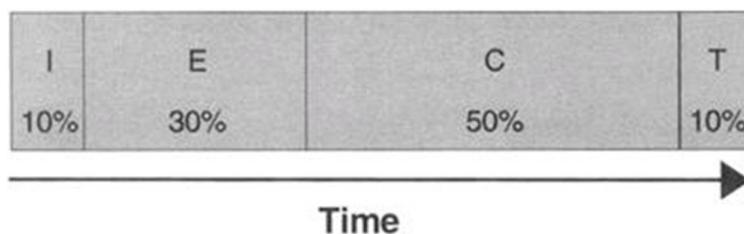
Giai đoạn Construction

- Đây là giai đoạn xây dựng sản phẩm và phát triển các phiên bản, kiến trúc, các kế hoạch cho đến khi đạt được phiên bản hoàn thiện nhất – có thể chuyển giao cho người dùng
- Giai đoạn này sẽ được kết luận dựa vào các mốc là khả năng thực hiện các chức năng yêu cầu ban đầu đã xác định

Giai đoạn Transition

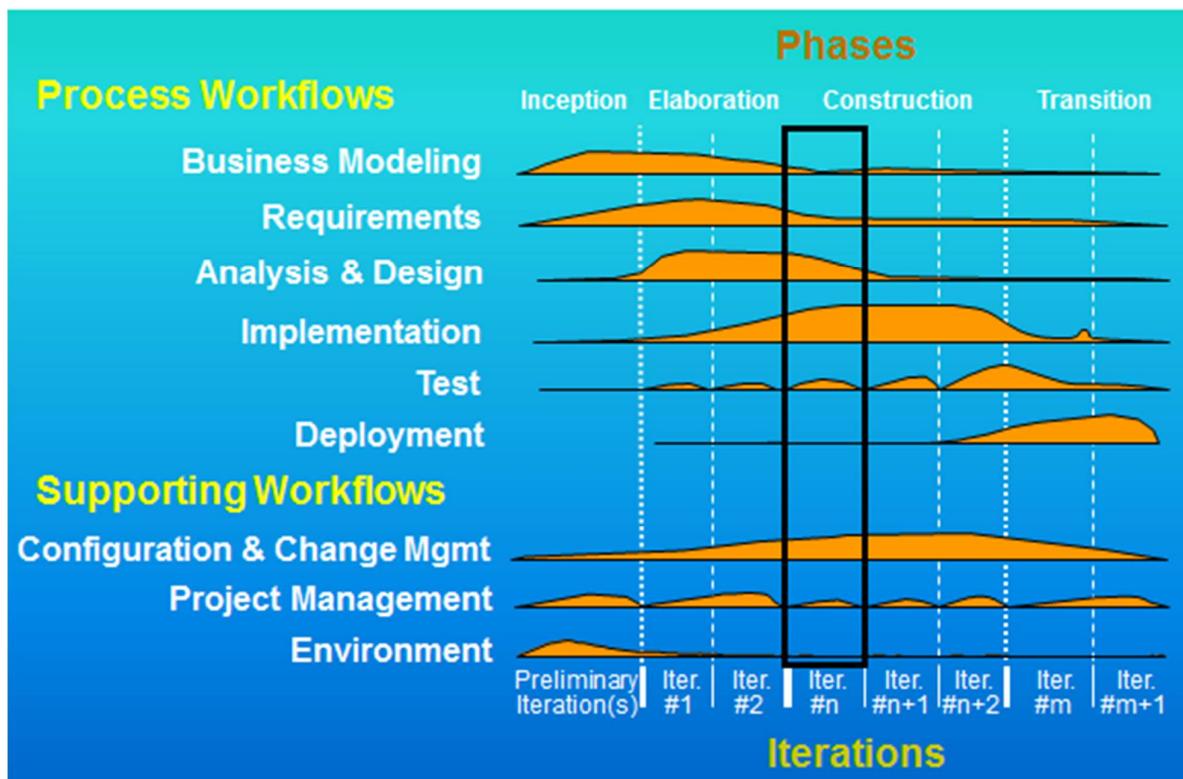
- Chuyển giao sản phẩm cho những người sử dụng nó, bao gồm: sản xuất, phân phát, huấn luyện, hỗ trợ và bảo trì cho đến khi người sử dụng hài lòng.
- Giai đoạn này được kết luận thông qua mốc các phiên bản của sản phẩm – kết thúc từng chu trình lặp của giai đoạn này

Thời gian dành cho các giai đoạn này được ước tính như sau:



1.2.2 QUY TRÌNH RUP

Quy trình RUP bao gồm 4 phases và 9 workflow, được thể hiện như sau:



- Hàng ngang của biểu đồ mô tả qui trình về mặt thời gian và vòng đời của một giai đoạn trong qui trình (khởi tạo, chi tiết hóa, xây dựng và chuyển giao)
- Hàng dọc của biểu đồ mô tả trạng thái tĩnh của qui trình, các giai đoạn của qui trình

1.2.3 PHÁT TRIỂN THEO MÔ HÌNH LẶP

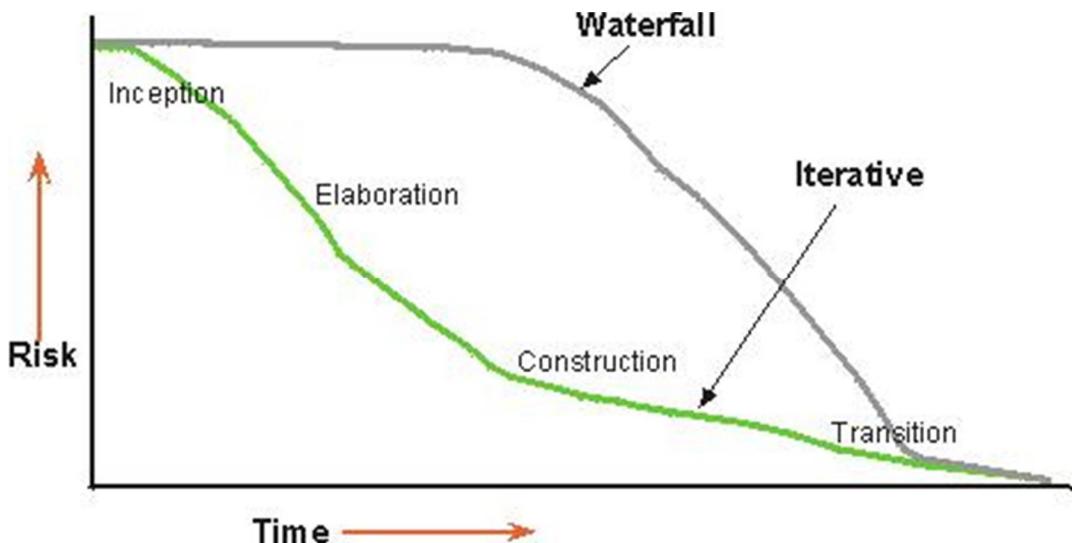
Một dự án sử dụng quy trình phát triển lặp lại có một vòng đời chứa các quá trình lặp lại. Một quá trình lặp là sự kết hợp chặt chẽ một chuỗi các hoạt động: mô hình nghiệp vụ, tiếp nhận yêu cầu, phân tích và thiết kế, thực thi, kiểm lỗi và triển khai với mức độ lặp không giống nhau, tùy theo vị trí cụ thể của vòng phát triển.

Quản lý, tiếp nhận yêu cầu và thiết kế là các hoạt động trọng điểm trong giai đoạn khởi tạo và phân tích chi tiết dự án; các hoạt động thiết kế, thực thi và kiểm lỗi đóng vai trò chủ chốt trong giai đoạn xây dựng; và các hoạt động kiểm lỗi, triển khai đóng vai trò chủ đạo trong giai đoạn chuyển giao dự án.



Một thiết kế ban đầu chỉ là một sản phẩm chưa hoàn thiện, dựa trên các yêu cầu căn bản, về sau quá trình thiết kế càng phát hiện ra thêm nhiều nhược điểm đó là kết quả trả giá từ việc chạy thử và trong một số trường hợp dự án phải hủy bỏ.

Tất cả các dự án đều có một tập các rủi ro phức tạp. Lúc ban đầu bạn có thể xác định để ngăn ngừa một vài rủi ro theo đúng như kế hoạch của mình. Tuy nhiên có rất nhiều rủi ro mà bạn không thể phát hiện ra một cách đơn giản, trớn tru cho đến khi bạn cố gắng tích hợp hệ thống. Bạn sẽ không bao giờ có thể dự đoán trước được tất cả các rủi ro khi không quan tâm đến kinh nghiệm của nhóm phát triển.

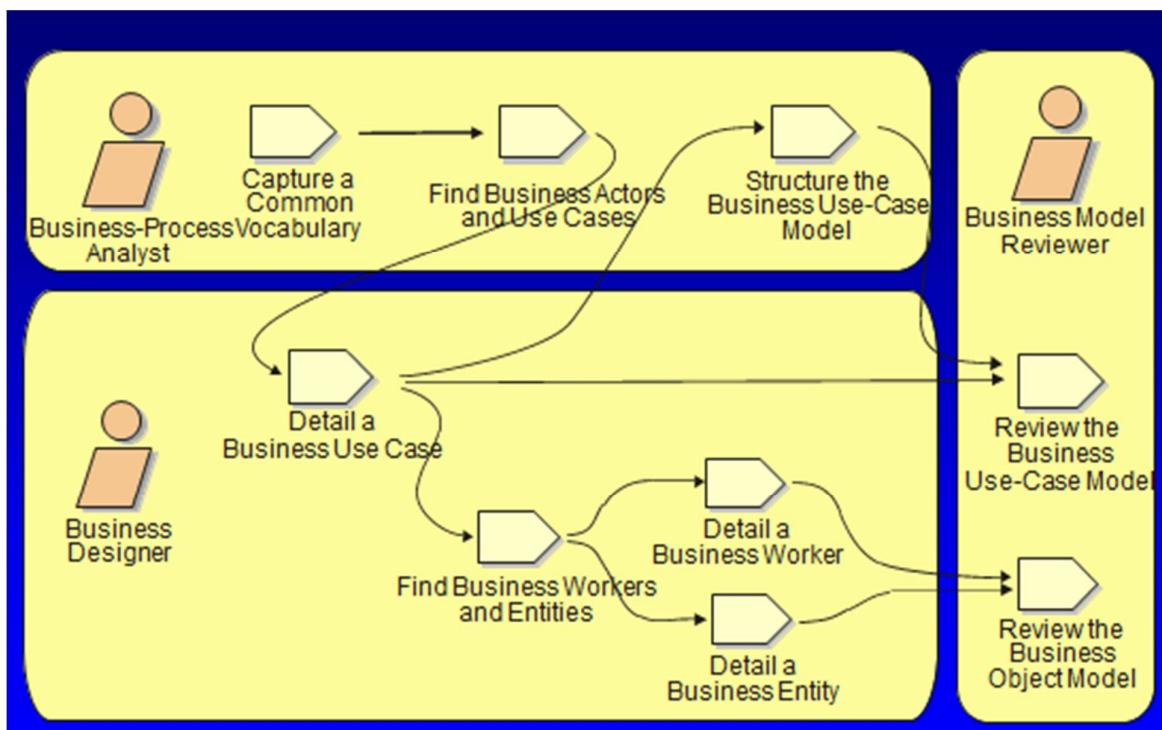


Lợi ích của phát triển lặp lại

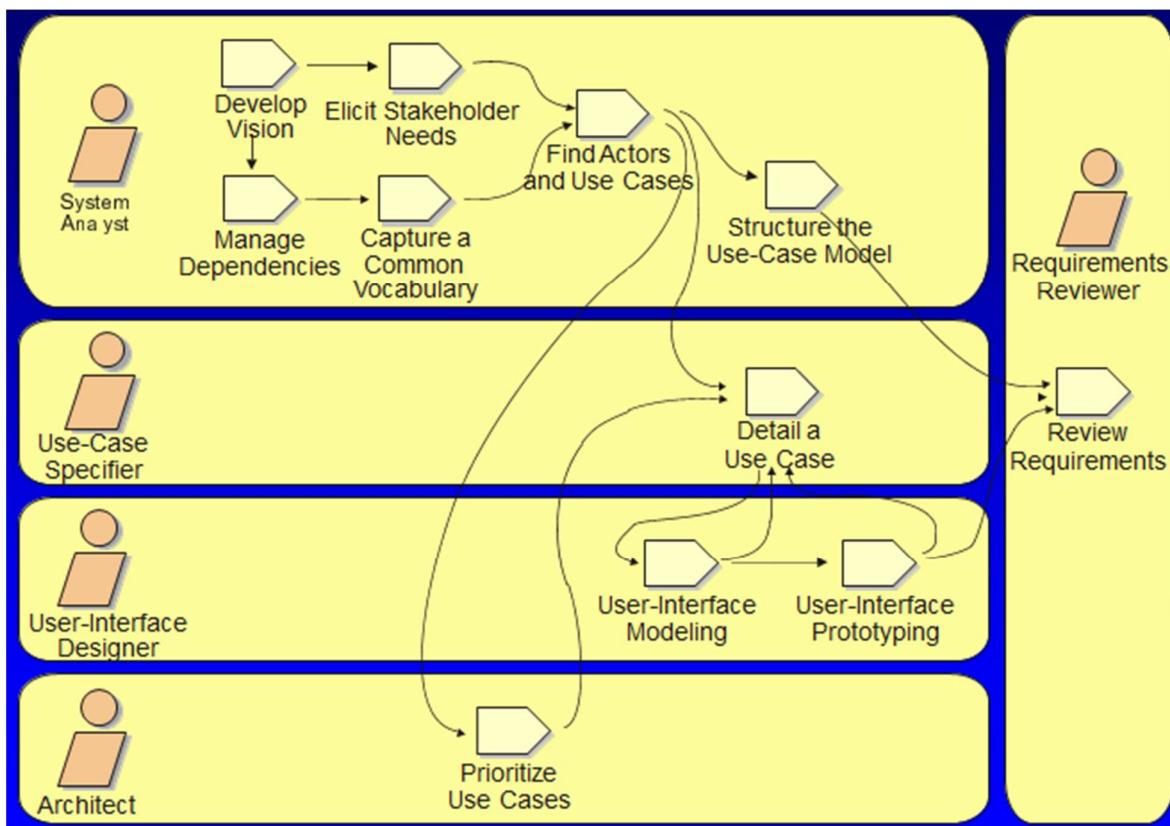
- Hạn chế được nhiều rủi ro do các phần tử được tích hợp, xây dựng dần dần
- Cho phép thay đổi các yêu cầu, các phương thức cho thích hợp hơn
- Các tổ chức có thể nắm được phương pháp này và phát triển cho qui trình của họ
- Tăng khả năng tái sử dụng

1.2.4 MỘT SỐ WORKFLOW CHÍNH TRONG RUP

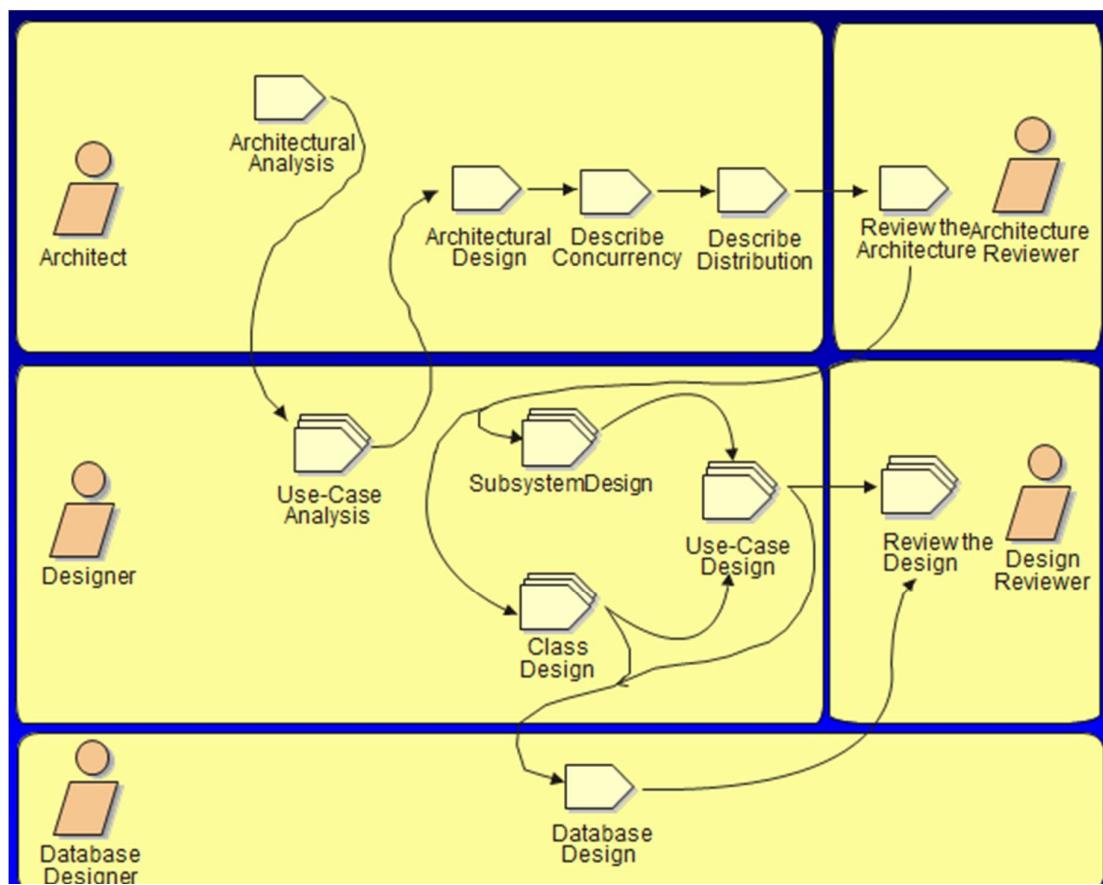
Mô hình hóa nghiệp vụ



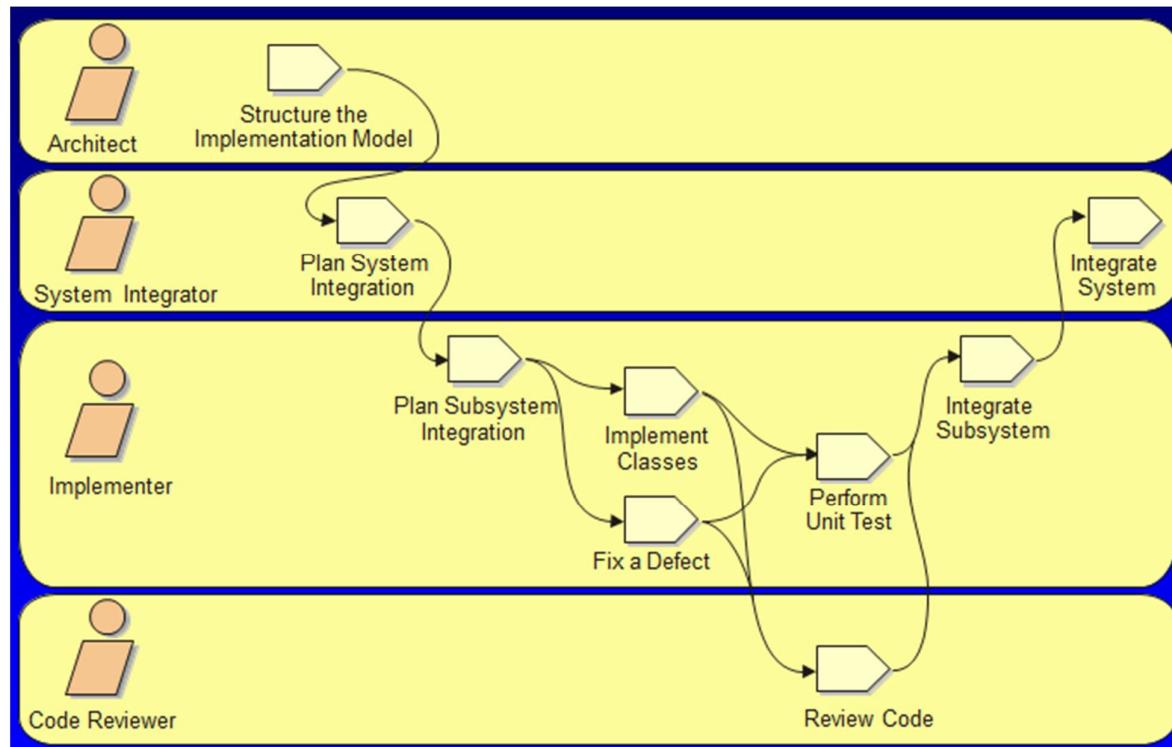
Lấy yêu cầu:



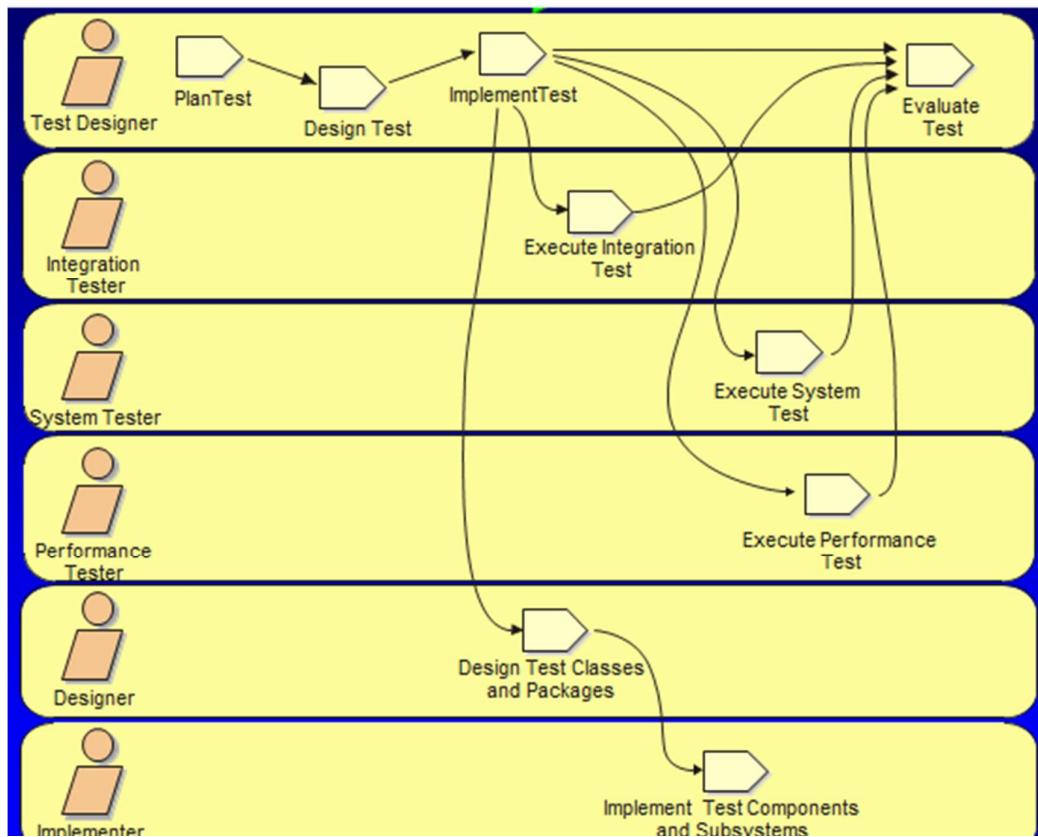
Phân tích thiết kế:



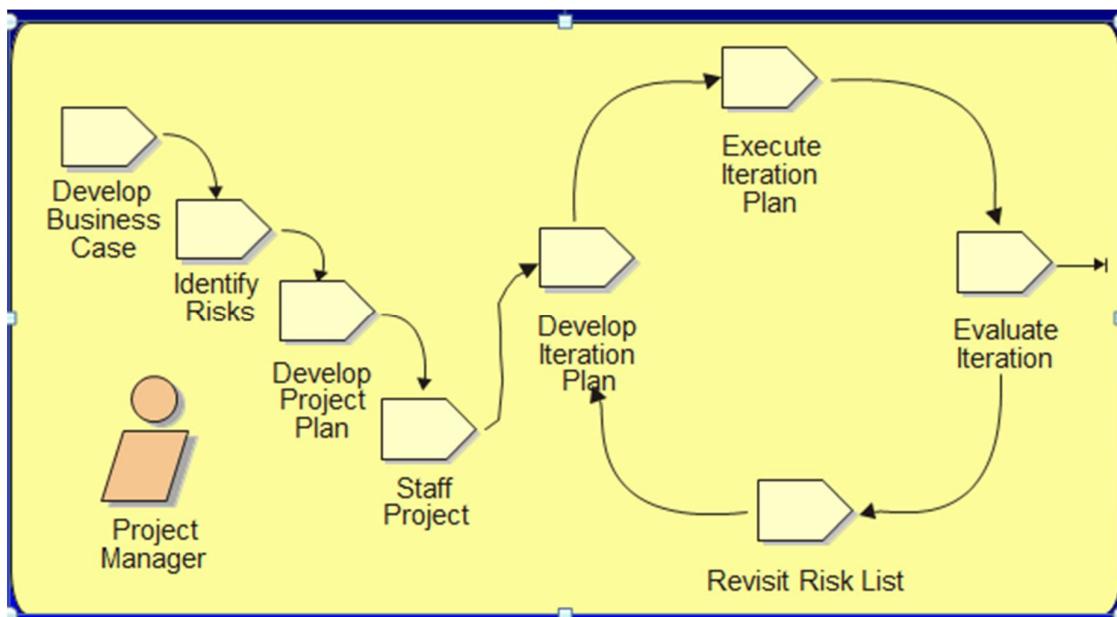
Hiện thực



Kiểm thử:



Quản trị dự án:



BÀI TẬP ĐỀ NGHỊ

Câu 1: Đọc và nghiên cứu quy trình phát triển phần mềm XP (eXtreme Programming)

BÀI 2: TỔNG QUAN UML

2.1 KHÁI NIỆM

Ngôn ngữ mô hình hóa hợp nhất (Unified Modeling Language - UML):

Là ngôn ngữ dùng để

- Xác định (Specifying)
- Trực quan hóa (Visualizing)
- Xây dựng (Constructing)
- Lập tài liệu (Documenting)



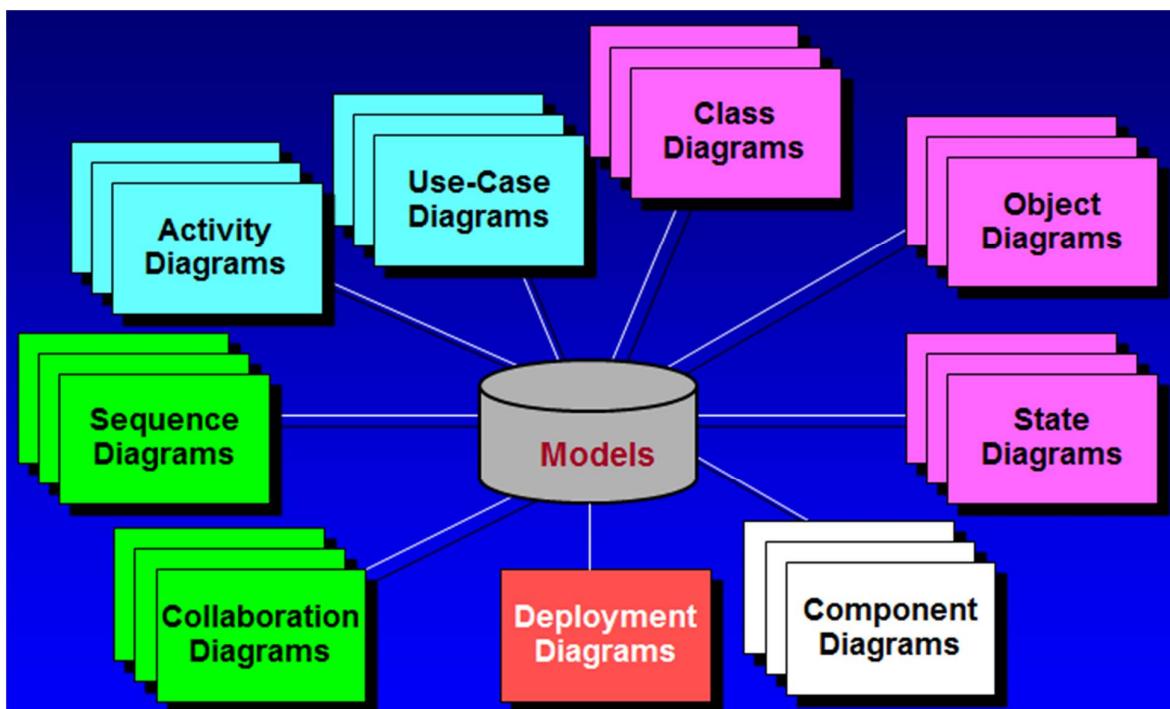
Cho các kết quả (artifacts) của quá trình thực hiện phần mềm.

Cách xây dựng các mô hình trong UML phù hợp mô tả các hệ thống thông tin cả về cấu trúc cũng như hoạt động. Cách tiếp cận theo mô hình của UML giúp ích rất nhiều cho những người thiết kế và thực hiện hệ thống thông tin cũng như những người sử dụng nó; tạo nên một cái nhìn bao quát và đầy đủ về hệ thống thông tin dự định xây dựng. Cách nhìn bao quát này giúp nắm bắt trọn vẹn các yêu cầu của người dùng; phục vụ từ giai đoạn phân tích đến việc thiết kế, thẩm định và kiểm tra sản phẩm ứng dụng công nghệ thông tin. Các mô hình hướng đối tượng được lập cũng là cơ sở cho việc ứng dụng các chương trình tự động sinh mã trong các ngôn ngữ lập trình hướng đối tượng, chẳng hạn như ngôn ngữ C++, Java, ... Phương pháp mô hình này rất hữu dụng trong lập trình hướng đối tượng. Các mô hình được sử dụng bao gồm Mô hình đối tượng (mô hình tĩnh) và Mô hình động.

UML sử dụng một hệ thống ký hiệu thống nhất biểu diễn các Phần tử mô hình (model elements). Tập hợp các phần tử mô hình tạo thành các Sơ đồ UML (UML diagrams). Có các loại sơ đồ UML chủ yếu sau:

- Sơ đồ tình huống sử dụng (Use Cases Diagram)
- Sơ đồ lớp (Class Diagram)
- Sơ đồ đối tượng (Object Diagram)
- Sơ đồ trình tự (Sequence Diagram)
- Sơ đồ cộng tác (Collaboration Diagram hay là Composite Structure Diagram)
- Sơ đồ trạng thái (State Machine Diagram)
- Sơ đồ thành phần (Component Diagram)
- Sơ đồ hoạt động (Activity Diagram)
- Sơ đồ triển khai (Deployment Diagram)
- Sơ đồ gói (Package Diagram)
- Sơ đồ liên lạc (Communication Diagram)
- Sơ đồ tương tác (Interaction Overview Diagram - UML 2.0)
- Sơ đồ phối hợp thời gian (Timing Diagram - UML 2.0)

UML ra đời do công của James Rumbaugh, Grady Booch và Ivar Jacobson.



2.2 CÁC BIỂU ĐỒ CỦA UML

2.2.1 BIỂU ĐỒ USECASE

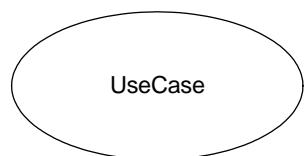
Một UC minh họa một đơn vị chức năng được hệ thống cung cấp. Mục đích chính của việc sử dụng sơ đồ UC là giúp các nhóm phát triển hình dung ra các yêu cầu chức năng của một hệ thống, bao gồm mối quan hệ của "các vai" (con người, người sẽ tương tác với hệ thống) với các quy trình cần thiết, cũng như các mối quan hệ trong số các UC khác nhau.

Actor



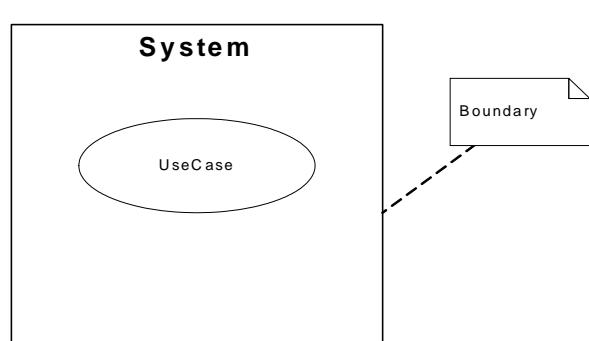
- Là người sử dụng hệ thống. Người sử dụng hệ thống có thể là người, máy, hệ thống khác hoặc một hệ thống con trong mô hình. Bất cứ tương tác nào từ bên ngoài hay bên trong hệ thống đều được gọi là Actor.

Use Case

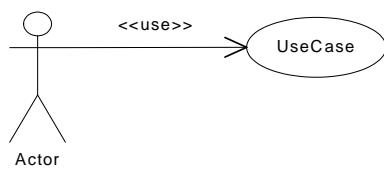


- Use case là một chức năng của hệ thống
- Use case là một kỹ thuật thu thập các yêu cầu chức năng của hệ thống
- Use case hoạt động dựa trên việc mô tả các tương tác đặc trưng giữa người sử dụng hệ thống và chính hệ thống (scenario)

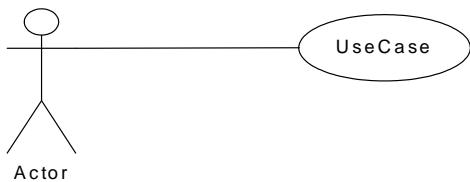
System Boundary



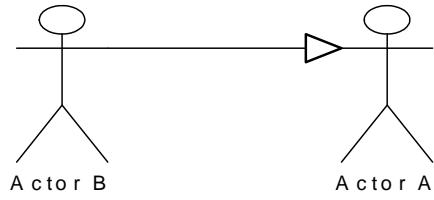
Là ranh giới giữa hệ thống và bên ngoài



- **Use:** Một actor hoặc usecase yêu cầu một actor hoặc usecase khác thực hiện một vài tương tác. Quan hệ use là một loại nhỏ hơn của quan hệ phục thuộc.

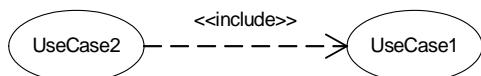


- **Association:** Thể hiện quan hệ giữa các thành phần trong mô hình.



của actor A

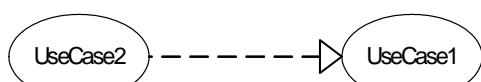
- **Generalization:** Thể hiện tính kế thừa, tính sử dụng lại của các actor
- Actor B thừa kế thuộc tính và vai trò



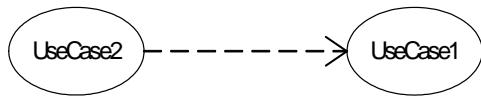
- **Inclusion:** Thể hiện UseCase2 bao gồm các chức năng của UseCase1



- **Extension:** UseCase2 mở rộng tác động, hành vi của UseCase1



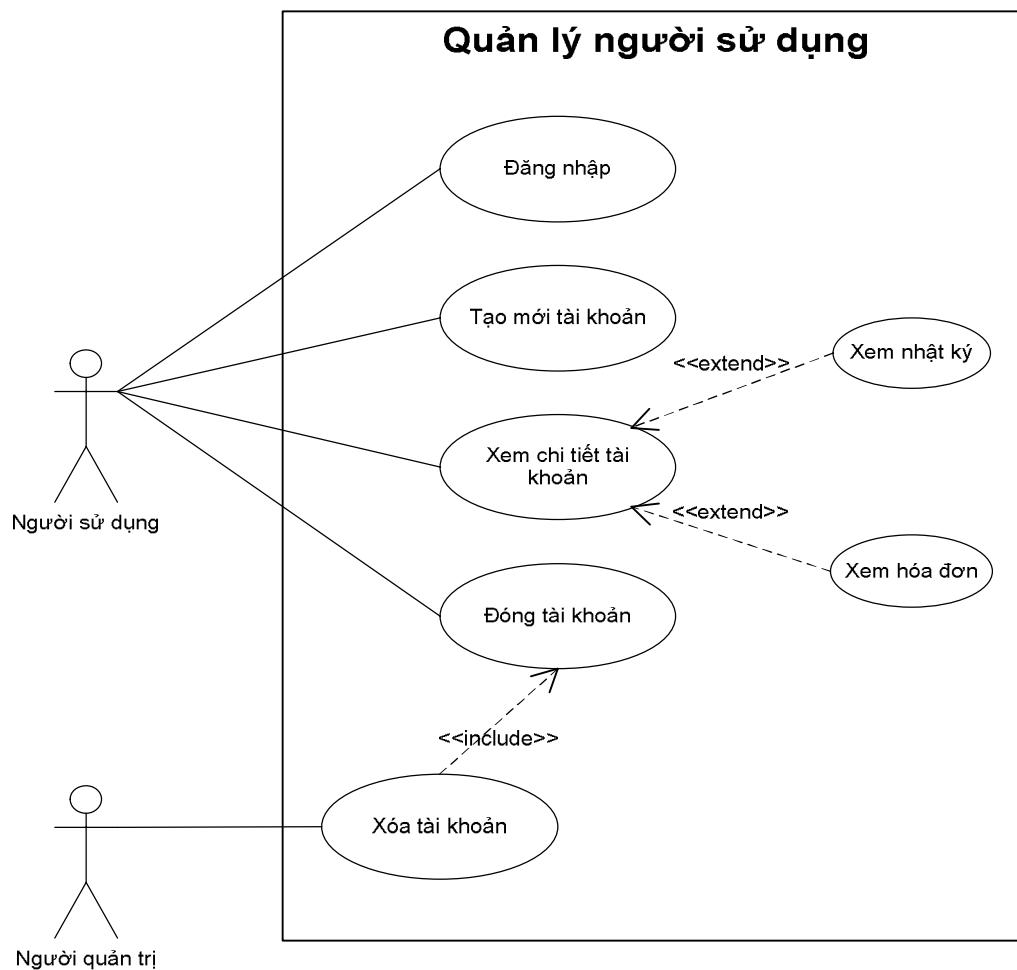
- **Realization:** UseCase2 thực hiện hoặc hiện thực hóa UseCase1



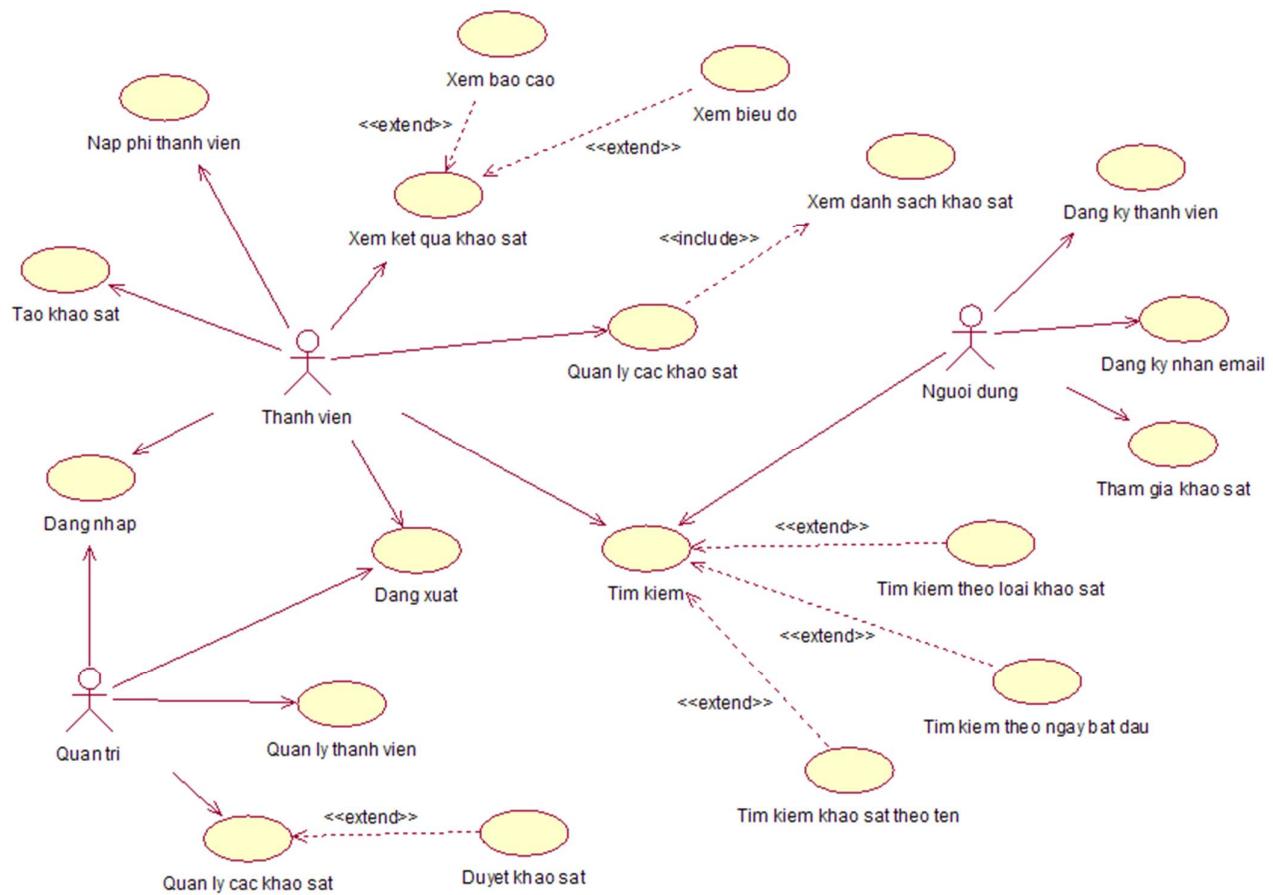
- **Dependency:** UseCase2 phụ thuộc vào UseCase1.

UseCase Diagram:

Một biểu đồ UseCase thể hiện các tương tác giữa các actor và các usecase. Nó thể hiện các yêu cầu chức năng của hệ thống, thể hiện sự tương tác giữa các tác nhân bên ngoài và bên trong hệ thống với hệ thống.



Bài tập: Đọc sơ đồ UC dưới đây:



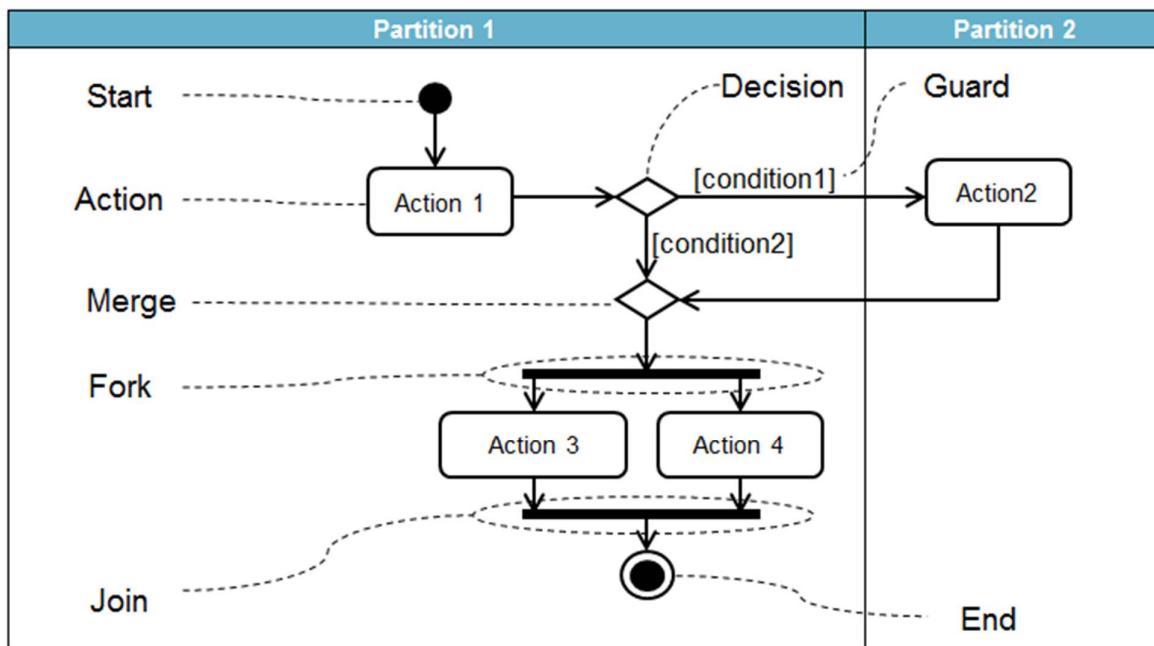
2.2.2 BIỂU ĐỒ HOẠT ĐỘNG – ACTIVITY DIAGRAM

Sơ đồ hoạt động hiển thị luồng kiểm soát theo thủ tục giữa hai hay nhiều đối tượng lớp khi xử lý một hoạt động.

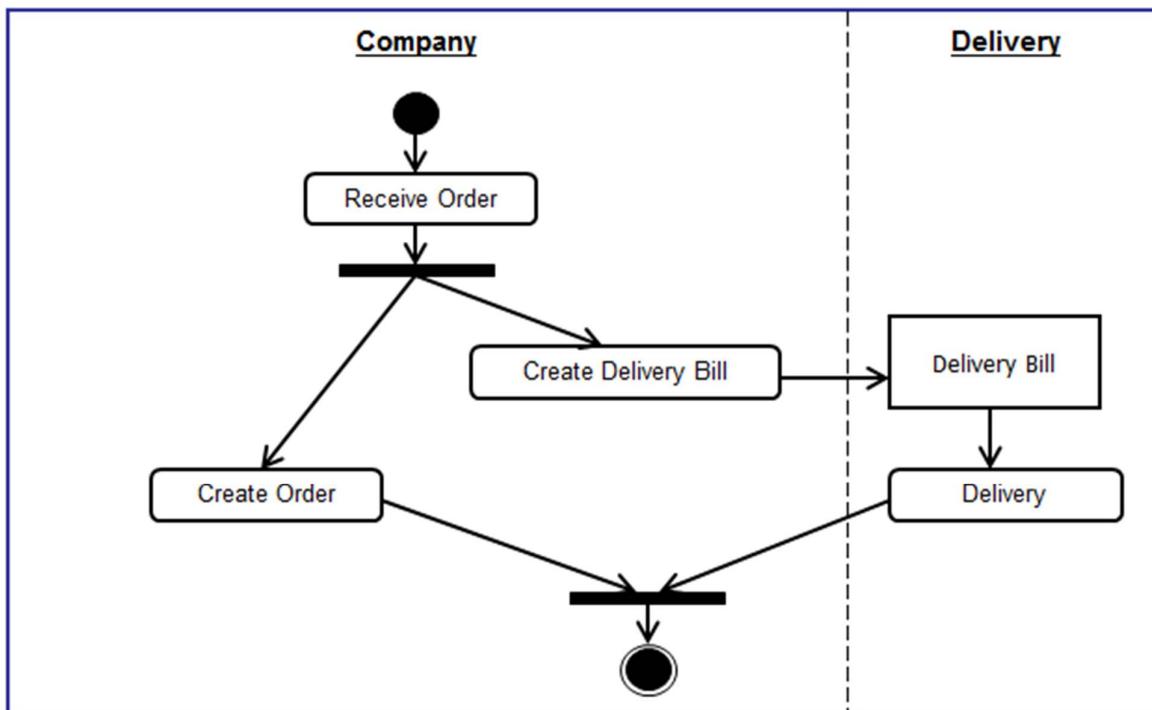
Mục đích của sơ đồ hoạt động

- Mô tả luồng nghiệp vụ của hệ thống
- Mô tả các hoạt động của UC
- Mô tả thuật toán

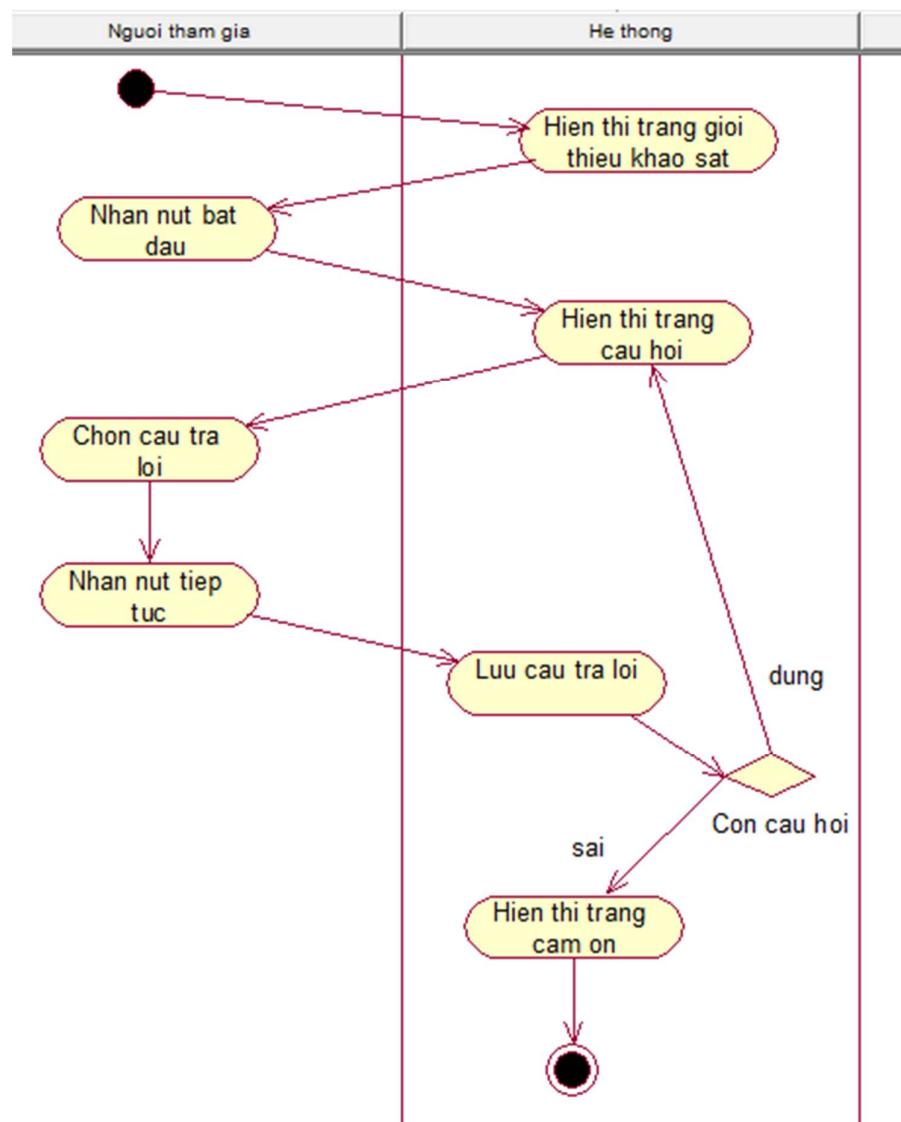
Các ký hiệu



Mô tả nghiệp vụ hệ thống



Mô tả hoạt động của UC



Mô tả thuật toán

Activity Diagrams Model Process Flow

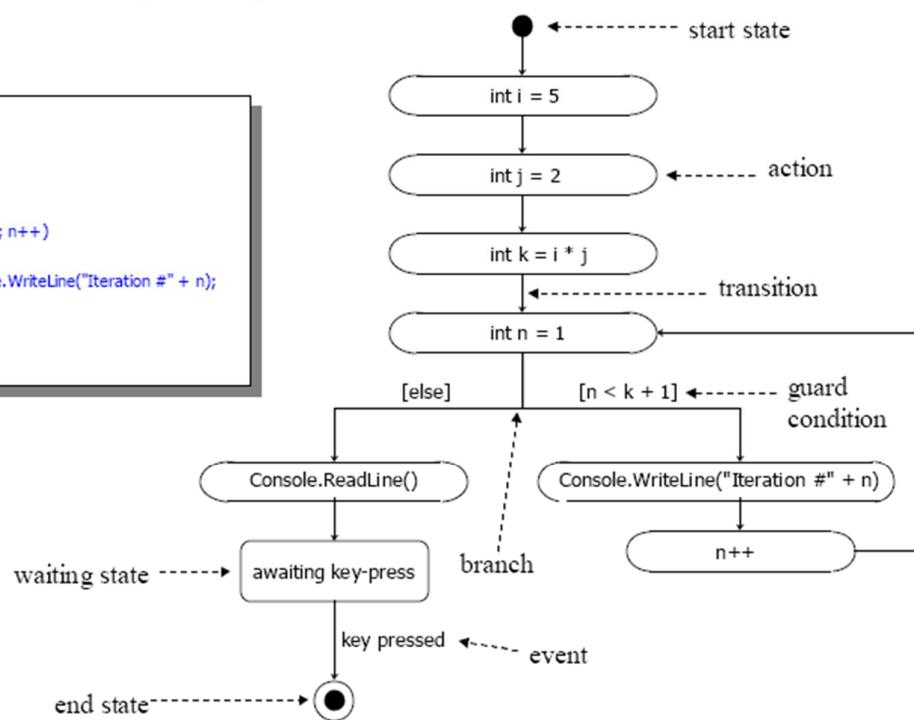
```

int i = 5;
int j = 2;
int k = i * j;

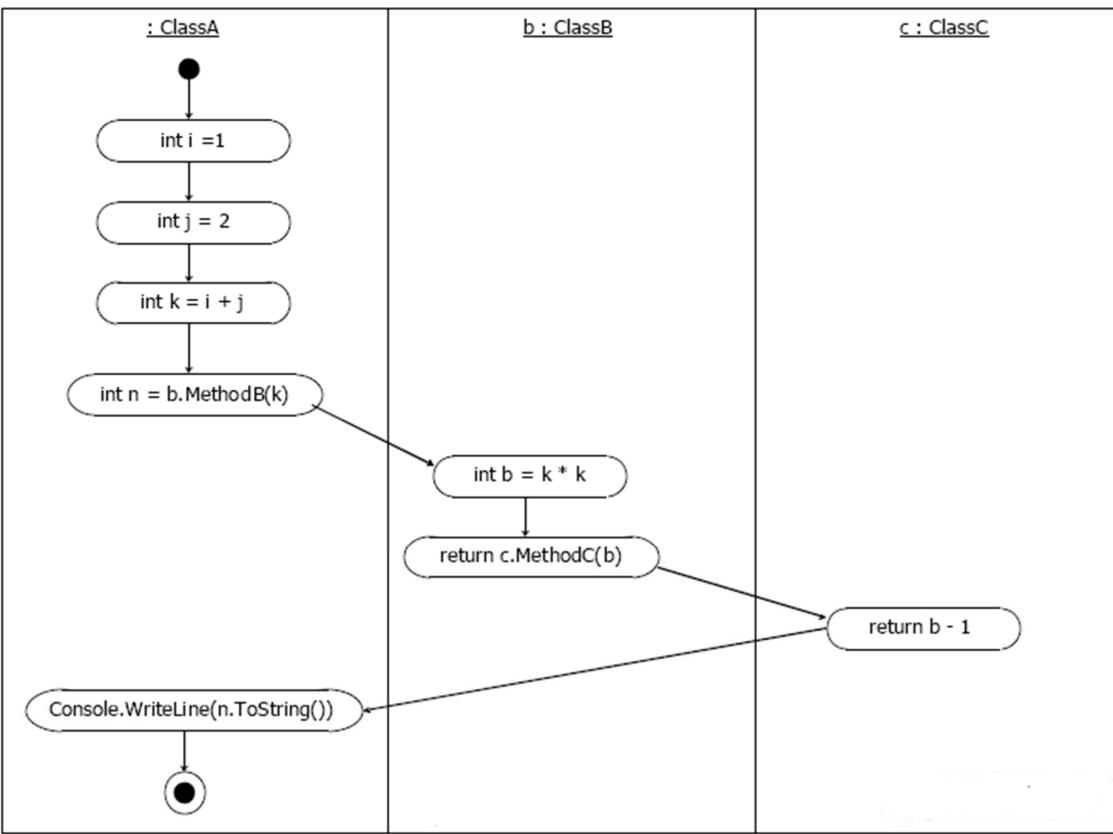
for(int n = 1; n < k + 1; n++)
{
    Console.WriteLine("Iteration #" + n);
}

Console.ReadLine();

```



Swim-lanes



2.2.3 BIỂU ĐỒ LỚP – CLASS DIAGRAM

Object:



- Trong thế giới thực chúng ta có thể các đối tượng là bất cứ gì như: Người, Xe, Cây, Bàn, Nhà....
- Trong lập trình thì các đối tượng thực chất là mô hình hóa của các đối tượng trong thế giới thực dưới mỗi ngôn ngữ lập trình hay mô hình khác nhau. Chúng ta có thể thấy các đối tượng đã được định nghĩa sẵn trong một số ngôn ngữ lập trình như Number, TextField, String, File, Windows, tiến trình, collection....
- Đối tượng không phải là kiểu dữ liệu trừu tượng, một đối tượng là một thể hiện của một class khi thực hiện. Ví dụ xe hơi có biển kiểm soát là "29A-1736" là một thể hiện của lớp "xe hơi" với thuộc tính là biển kiểm soát.
- Một Object bao giờ cũng có thuộc tính và phương thức. Thuộc tính là những gì mà object có sở hữu, và nó là kiến thức mà chính bản thân object có được. Phương thức, services là những gì mà object có khả năng làm được.

Class:



- Lớp là sự đại diện cho một tập các đối tượng có chung thuộc tính, phương thức. Class bao giờ cũng có thuộc tính (dữ liệu) và phương thức (dịch vụ, hành vi).



- **Association:** Thể hiện mối quan hệ giữa 2 lớp cũng như giữa 2 đối tượng (Link).

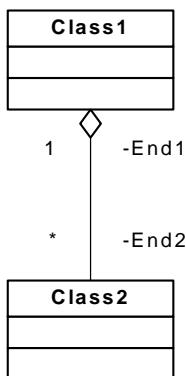


- Trong UML, sự kết hợp được định nghĩa là một quan hệ gồm một tập các mối liên kết, mỗi liên kết là một liên kết ngữ nghĩa giữa hai đối tượng hay mối quan hệ giữa hai lớp.



- Có 2 kiểu quan hệ kết hợp là một

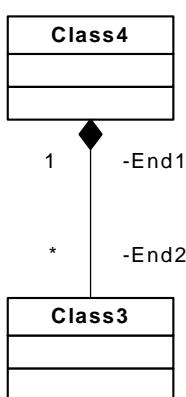
phía (uni-direction) và hai phía (bi-direction)



- **Aggregation:** là một trường hợp đặc biệt của quan hệ kết hợp được dùng để biểu diễn "Tổng thể - Thành phần", điều đó có nghĩa là một lớp sẽ bao gồm một hoặc nhiều lớp khác.

- Với quan hệ tập hợp thì khi Class1 bao gồm Class2 nhưng không sở hữu. Class2 tồn tại một cách độc lập, khi Class1 mất đi thì Class2 không bị mất đi.

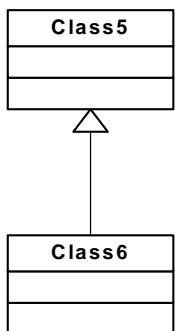
- Quan hệ aggregation được coi như quan hệ có môt(has - a).



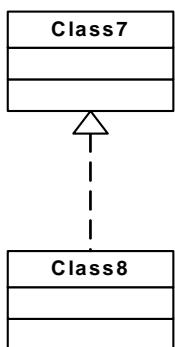
- **Composition:** Mạnh hơn aggregation ở chỗ Class4 bao gồm Class3 nhưng lại sở hữu Class3. Class3 không tồn tại bên ngoài Class4 và khi Class4 mất đi thì Class3 cũng mất đi.

- Với quan hệ tập hợp thì khi Class tạo thành mất đi(Class1) thì Class được tạo thành không bị mất đi(Class2).

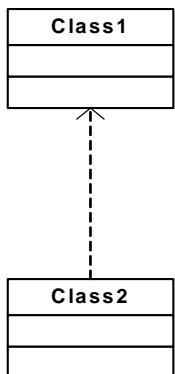
- Quan hệ aggregation được coi như quan hệ có môt(has - a).



- **Generalization:** Là quan hệ giữa một lớp tổng quát (Class5) và một lớp đặc biệt (Class6).
- Class5 được gọi là lớp cha và Class6 được gọi là lớp con.
- Lớp con được kế thừa toàn bộ thuộc tính và phương thức mà lớp cha có.



- **Realization:** Class8 hiện thực hóa Class7. Class8 là lớp hiện thực và Class7 là lớp đặc tả.
- Class7 là Interface và Class8 là một lớp thực hiện Interface

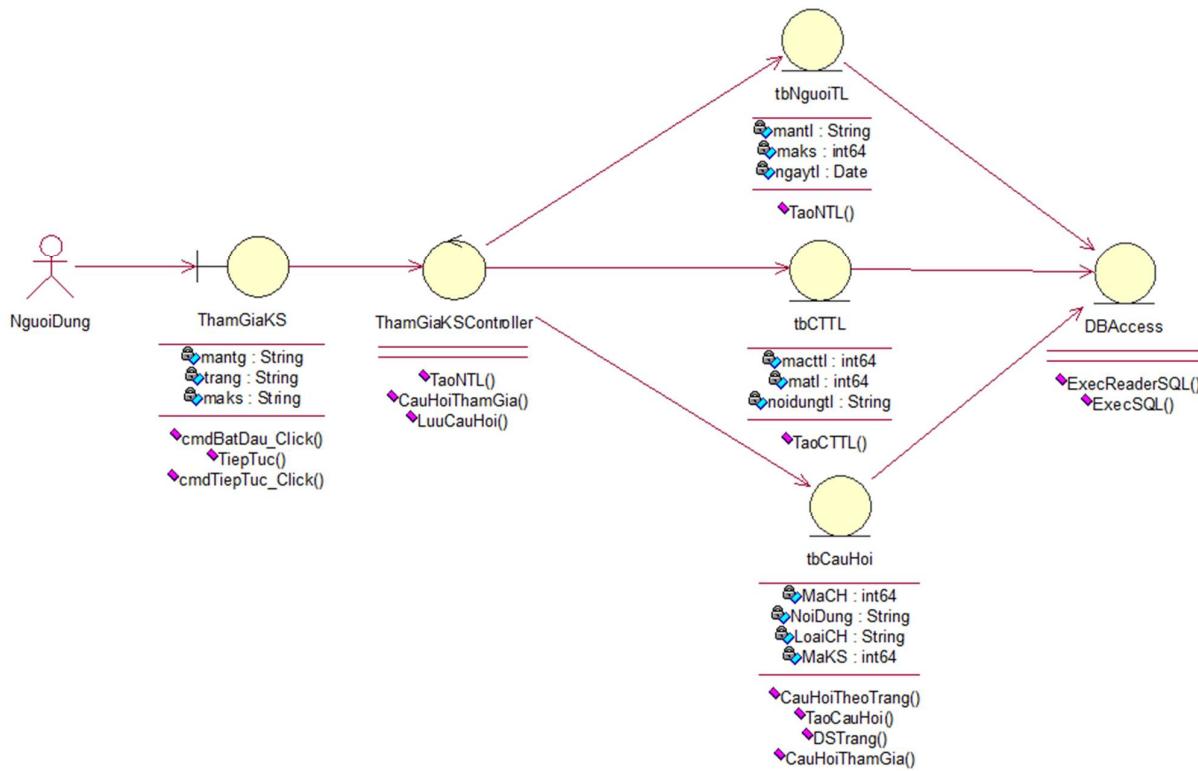


- **Dependency:** Là một liên kết giữa 2 lớp trong đó một lớp độc lập(Class1) và một lớp phụ thuộc(Class2). Những thay đổi trong lớp độc lập sẽ ảnh hưởng đến lớp phụ thuộc.

- **Visibility (Scope):** Thể hiện phạm vi truy cập đối với thuộc tính hay phương thức của Class...
- **Stereotype:** có nghĩa là làm gia tăng thêm ý nghĩa của một Class. Nội dung được đặt trong <<abc>>, ở đây abc chỉ mang tính chất giải thích rõ nghĩa thêm mà không có ảnh hưởng gì tới Class.
- **Cardinality (Multiplicity):** Xác định số đối tượng của mỗi lớp có thể liên kết với nhau. Và nó có thể là: *, 0, 0..*, 0..1, 1, 1.., 1..*, 2..8
- Thông thường, người ta phân loại class thành 3 loại đặc biệt sau:

- Boundary Class – Lớp biên:** Là một lớp để mô tả việc tương tác giữa actor với hệ thống, điển hình ta có thể thấy Boundary là giao diện chương trình. Nó thể hiện các tương tác giữa người dùng với hệ thống ở mức độ giao diện màn hình.
- Entity Class – Lớp dữ liệu:** Là một kho (Store) để thu thập thông tin và knowledge trong hệ thống.
- Controller Class – Lớp điều khiển:** Là một khuôn lớp để thể hiện các control, quản lý các thực thể. Control được tổ chức và schedule cho các tương tác khác nhau với các thành phần khác nhau.

Class diagram

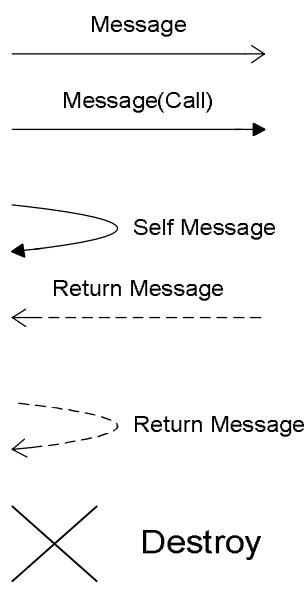


2.2.4 BIỂU ĐỒ TUẦN TỰ - SEQUENCE DIAGRAM

Biểu đồ tuần tự thể hiện sự tương tác giữa hai hay nhiều object để hiện thực các chứng năng của hệ thống theo thứ tự về thời gian. Nó được sử dụng để mô tả dòng thôna điệp được gửi đi và và các đổi tượna phổi hơp nhân và

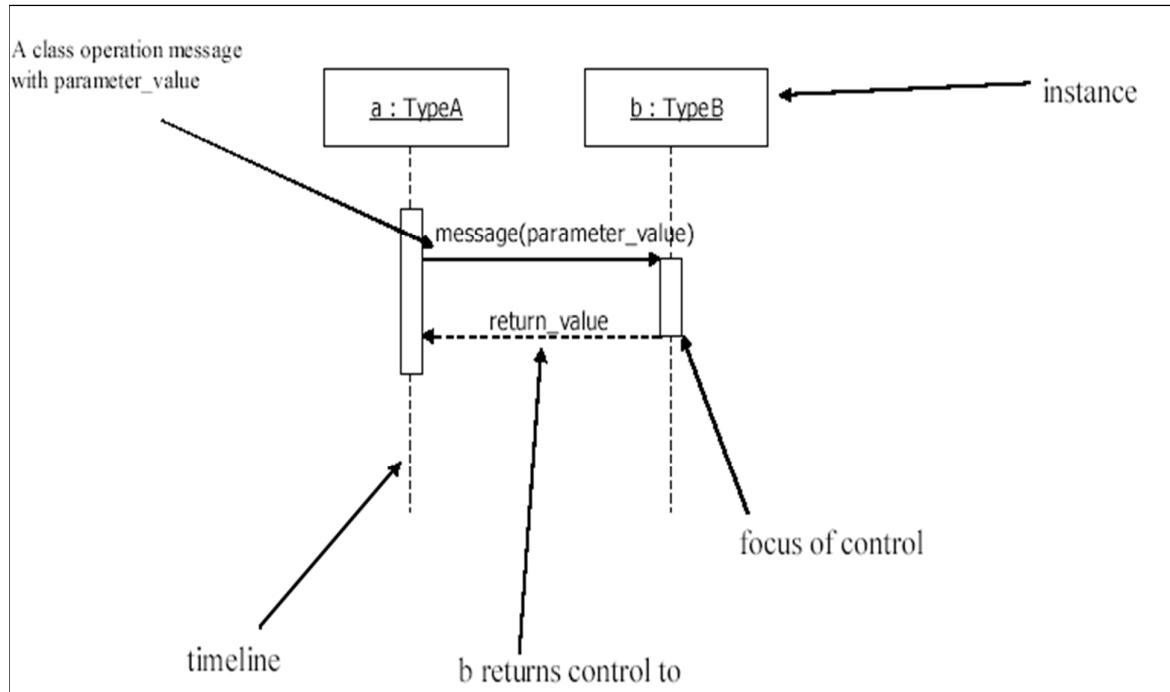
Biểu đồ tuần tự thông thường được sử dụng như một mô hình giải thích cho kịch bản usecase. Biểu đồ tuần tự thể hiện rất rõ đối tượng nào tương tác với đối tượng nào và thông điệp là gì. Khi đọc một biểu đồ tuần tự ta đọc từ trái qua phải và từ trên xuống dưới.

- **Lifelines:** Thể hiện sự tồn tại của đối tượng theo thời gian (thời gian sống của đối tượng). Trong UML nó được biểu diễn bởi đường nét rời đứng.
- **Activations:** Thể hiện thời gian, chu trình sống của đối tượng khi thực hiện một service nào đó (thời gian mà service đó còn tồn tại). Trong UML nó được biểu diễn bằng một hình chữ nhật hép, đứng.

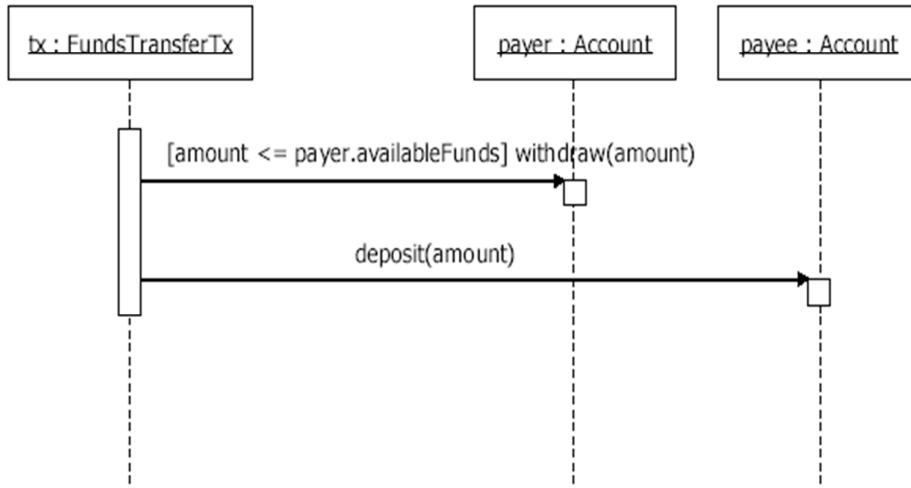


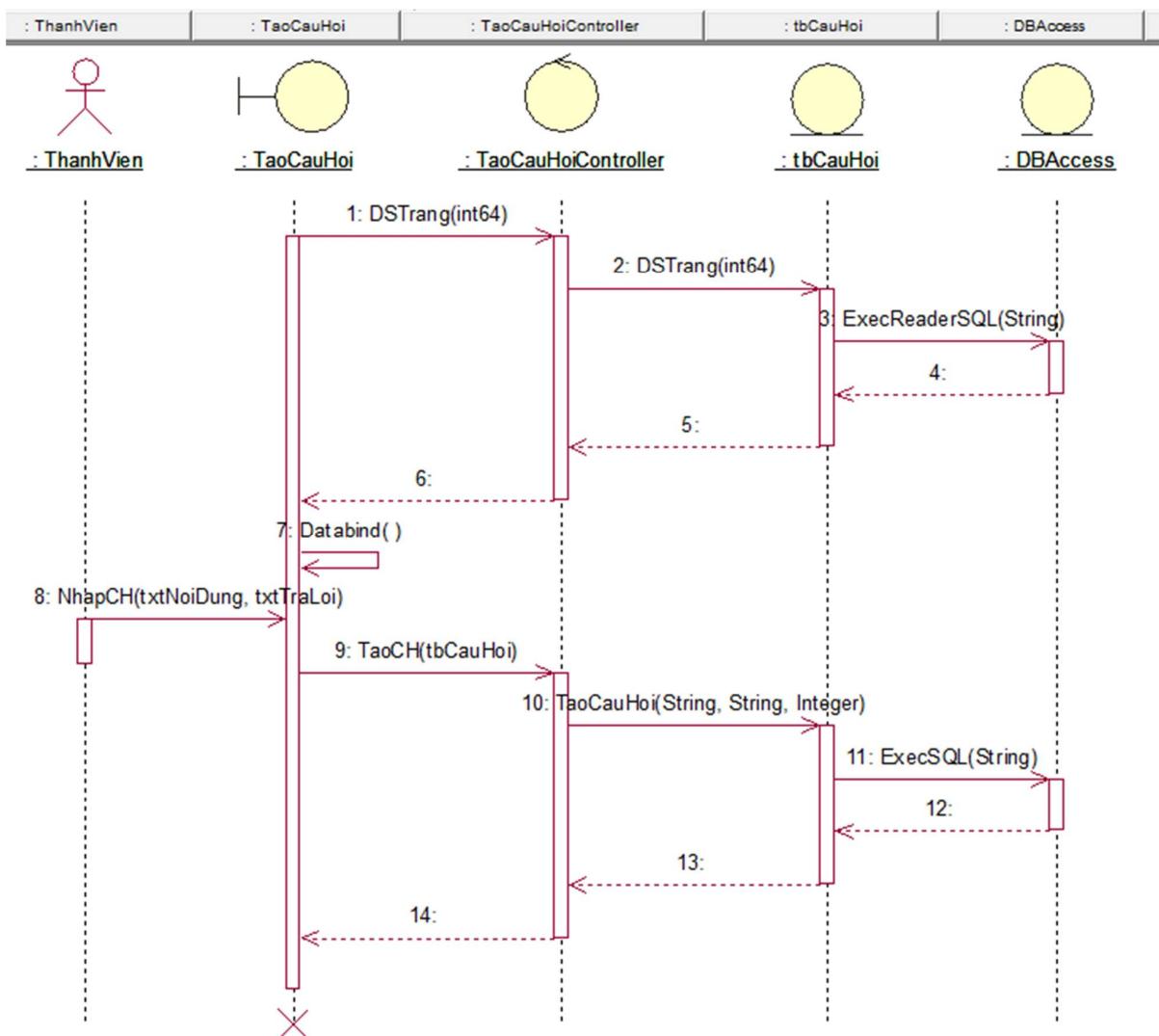
- **Message:** Một object gửi một thông điệp đến một object khác nhờ thực hiện một service nào đó mà nó không có. Khi gửi một message có thể có tham số kèm theo và đó là knowledge (data) của object gửi.
- **Message (Call, Procedure):** gọi một phương thức cụ thể của object đích.
- **Self message:** Self message là tự gọi một phương thức ngay tại trong lớp đó.
- **Return message:** Là message kết quả trả về sau khi đã thực hiện một message gửi
- **Destroy:** Kết thúc chu trình sống của đối tượng thì ta phải huỷ đối tượng.

Sơ đồ tuần tự



Conditional Messages

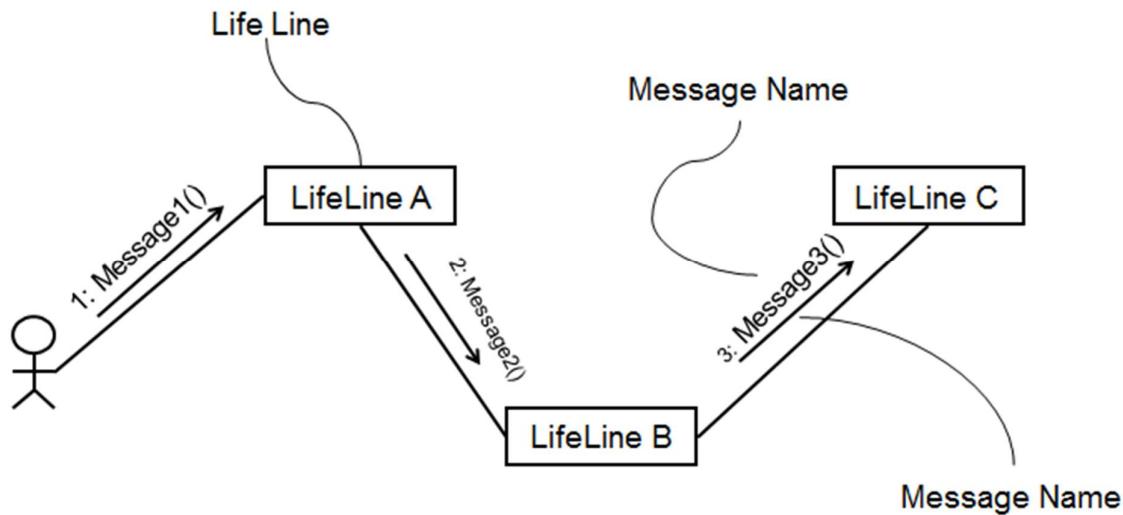




2.2.5 BIỂU ĐỒ GIAO TIẾP – COMMUNICATION DIAGRAM

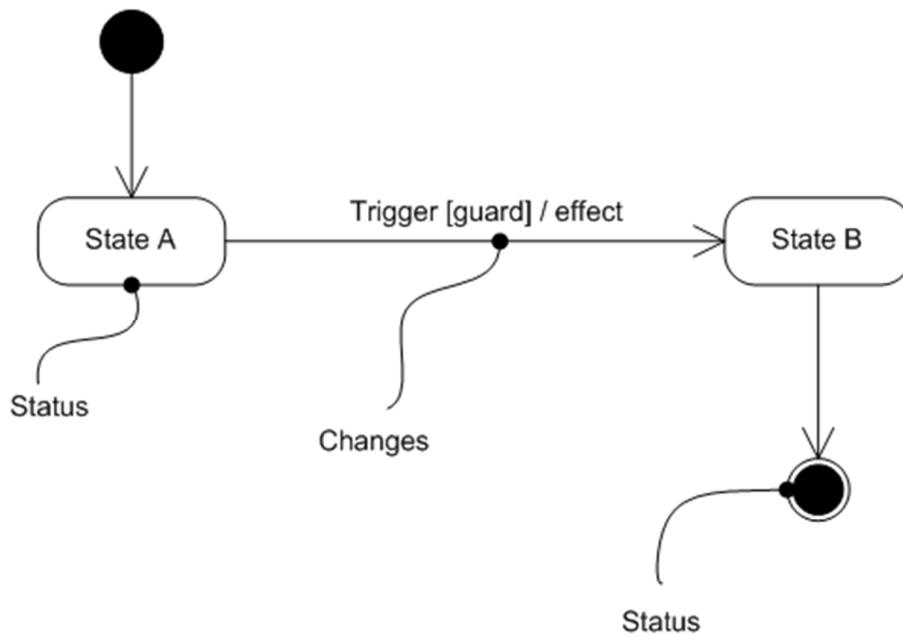
Một biểu đồ giao tiếp miêu tả tương tác giữa các đối tượng cũng giống như biểu đồ tuần tự, nhưng nó tập trung trước hết vào các sự kiện, tức là tập trung chủ yếu vào sự tương tác giữa các đối tượng.

Trong một biểu đồ cộng tác, các đối tượng được biểu diễn bằng kí hiệu lớp. Thứ tự trong biểu đồ cộng tác được thể hiện bằng cách đánh số các thông điệp. Kỹ thuật đánh số được coi là hơi có phần khó hiểu hơn so với kỹ thuật mũi tên sử dụng trong biểu đồ tuần tự.

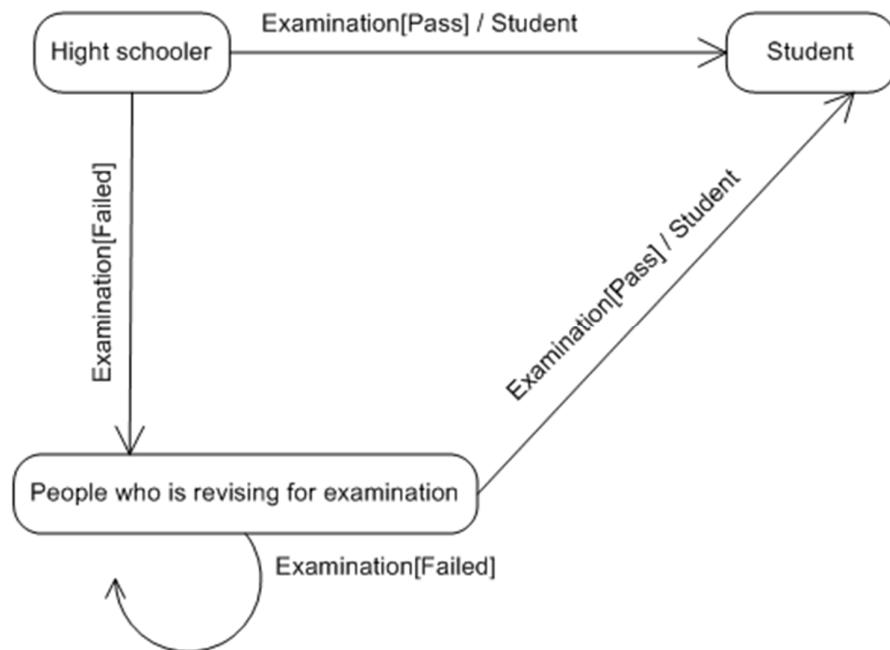


2.2.6 BIỂU ĐỒ TRẠNG THÁI – STATE DIAGRAM

Biểu đồ trạng thái thể hiện vòng đời của các đối tượng, các hệ thống con (Subsystem) và các hệ thống. Chúng cho ta biết các trạng thái mà một đối tượng có thể có và các sự kiện (các thông điệp nhận được, các khoảng thời gian đã qua đi, các lỗi xảy ra, các điều kiện được thỏa mãn) sẽ ảnh hưởng đến những trạng thái đó như thế nào dọc theo tiến trình thời gian.

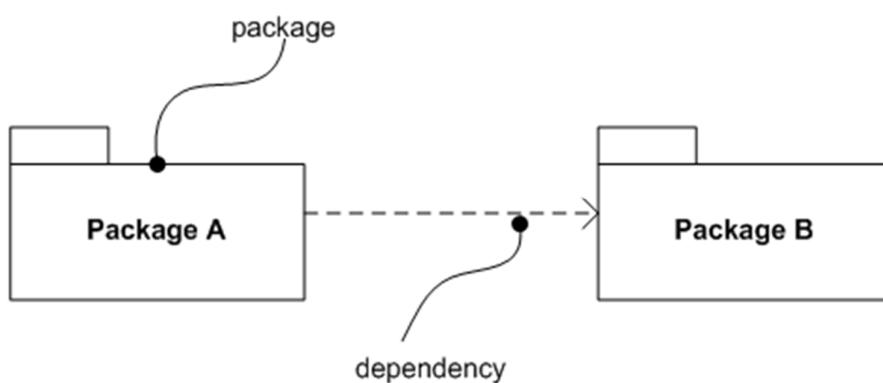


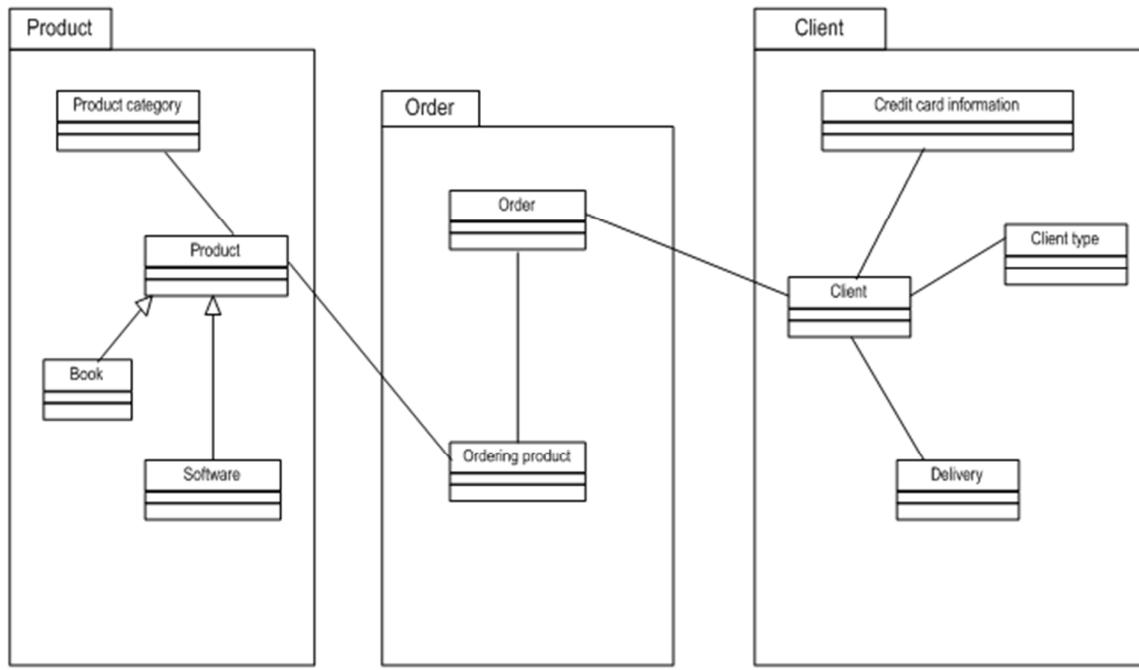
Ví dụ biểu đồ trạng thái:



2.2.7 BIỂU ĐỒ GÓI – PACKAGE DIAGRAM

- Package dùng để nhóm các phần tử trong UML lại với nhau theo một nghĩa nhất định, với mục đích làm đơn giản hóa quá trình tổ chức, quản lý các thành phần trong UML.





2.2.8 BIỂU ĐỒ THÀNH PHẦN – COMPONENT DIAGRAM

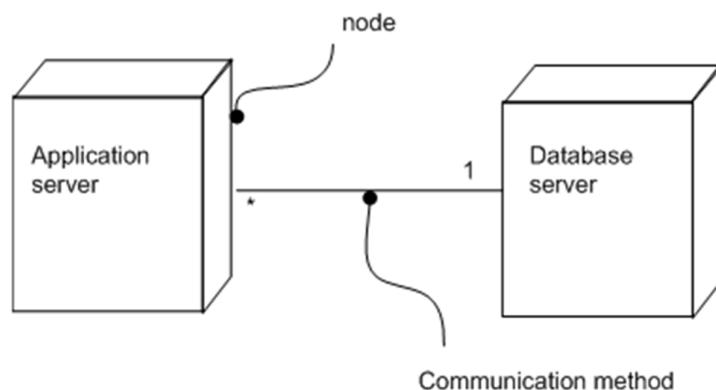
Một biểu đồ thành phần cung cấp một khung nhìn vật lý của hệ thống. Mục đích của nó là hiển thị các phụ thuộc mà phần mềm có trên các thành phần phần mềm khác (ví dụ, các thư viện phần mềm) trong hệ thống. Sơ đồ này có thể được hiển thị ở mức rất cao, chỉ với các thành phần có độ chi tiết lớn hoặc nó có thể được hiển thị tại mức gói.



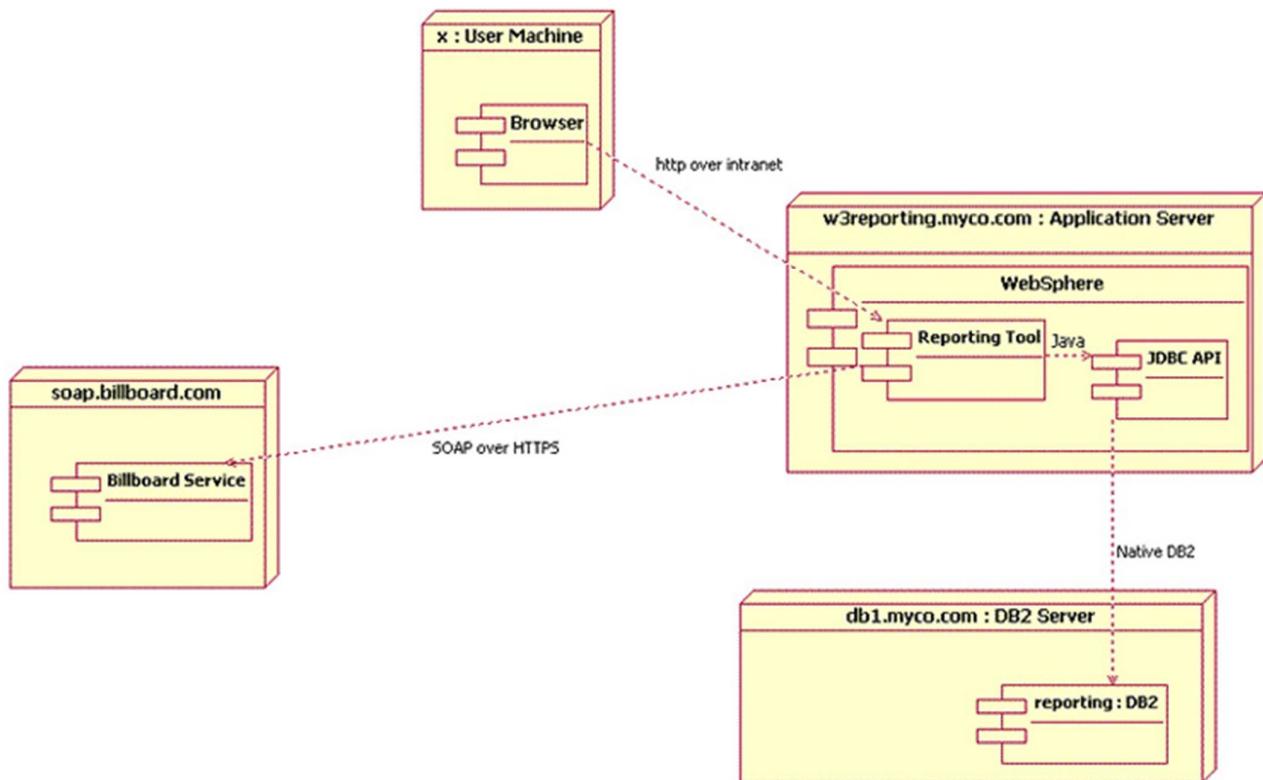
2.2.9 BIỂU ĐỒ TRIỂN KHAI – DEPLOYMENT DIAGRAM

Biểu đồ triển khai cho thấy cách một hệ thống sẽ được triển khai cụ thể

nhau của hệ thống cụ thể sẽ chạy ở đâu và làm thế nào để chúng giao tiếp với nhau.



Ví dụ:



BÀI TẬP ĐỀ NGHỊ

Công ty Skydoor muốn phát triển một hệ thống web để quảng bá du lịch tại Việt Nam. Trang web có tích hợp google map để hiển thị bản đồ của các địa điểm du lịch. Hệ thống có các yêu cầu như sau:

Đối với mọi người sử dụng web, trang web phải hiển thị được bản đồ của nước Việt Nam với các tỉnh thành và địa điểm du lịch cho các tỉnh thành đó. Người sử dụng có thể đọc các bài mô tả về các địa điểm, tìm kiếm thông tin về các địa điểm theo tỉnh thành, theo tên địa điểm. Người sử dụng có thể đăng ký thành viên của trang web.

Đối với thành viên, trang web phải cung cấp các chức năng cho phép thêm địa điểm mới, viết bài mô tả, hoặc đánh giá cho từng địa điểm. Thành viên có thể tự thiết kế các tour du lịch cho riêng mình theo các địa điểm hiện có. Sau đó hệ thống có thể cho phép thành viên in ra chi tiết tour du lịch mà mình tự thiết kế.

Đối với người quản trị web, người quản lý có quyền quản lý các thành viên, quản lý các bài đánh giá, các địa điểm du lịch do các thành viên đăng lên. Người quản lý cũng có quyền chọn để xuất bản các địa điểm do thành viên thêm vào để chia sẻ với các thành viên khác trong cộng đồng, nếu địa điểm chưa được người quản trị cho phép xuất bản thì địa điểm này chỉ thấy được bởi người tạo ra nó. Hệ thống cũng có các thống kê cho biết địa điểm du lịch nào được nhiều người đến nhất.

Hãy phân tích và thiết kế hệ thống trên theo yêu cầu sau:

1. Hãy vẽ Usecase diagram cho hệ thống trên.
2. Hãy vẽ Entity class diagram cho hệ thống trên.
3. Hãy thiết kế giao diện cho usecase thành viên thiết kế tour du lịch cho mình.
4. Hãy vẽ collaboration diagram cho usecase thành viên viên thiết kế tour du lịch cho mình.

5. Vẽ sơ đồ triển khai của hệ thống này.

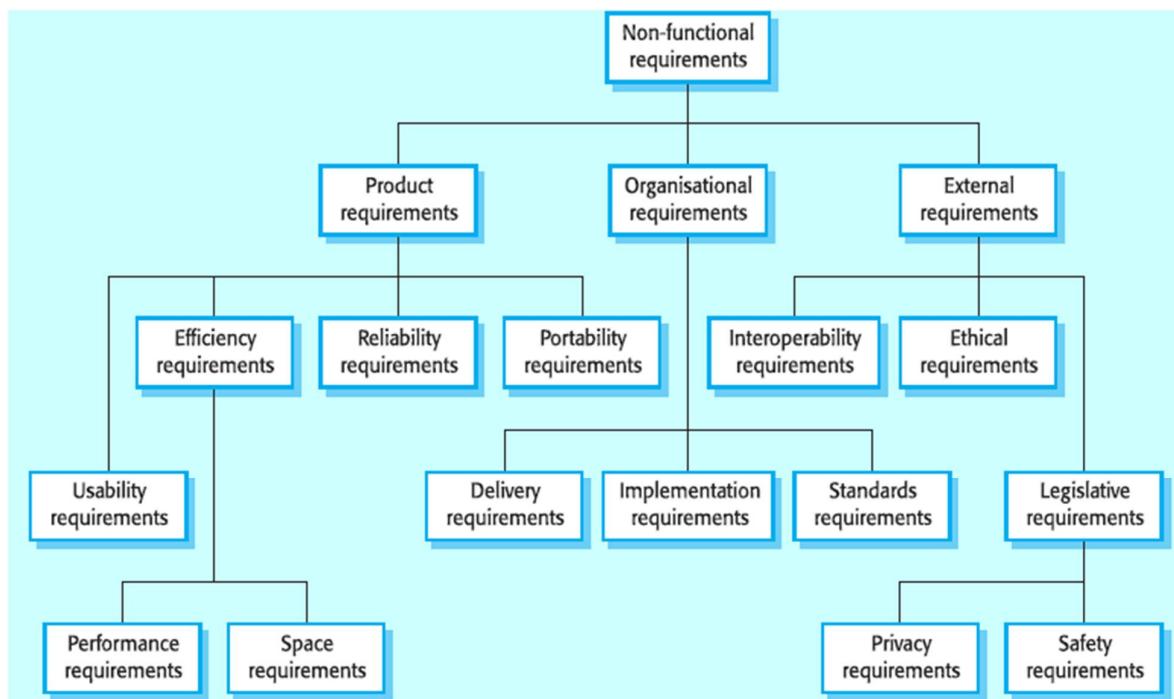
BÀI 3: LẤY YÊU CẦU

3.1 KHÁI NIỆM

Một yêu cầu là một đặc trưng của hệ thống, hay là sự mô tả những việc, mà hệ thống có khả năng thực hiện để hoàn thành mục tiêu của hệ thống.

Xác định yêu cầu: là mô tả các dịch vụ mà hệ thống được mong đợi phải cung cấp và các ràng buộc mà hệ thống phải tuân thủ khi vận hành. Nó chỉ có các đặc tả bên ngoài của hệ thống mà không liên quan đến các đặc tính thiết kế. Nó phải được viết sao cho người ta có thể hiểu được mà không cần một kiến thức chuyên môn đặc biệt nào. Các yêu cầu của phần mềm được chia thành hai loại:

- **Các yêu cầu hệ thống chức năng:** các dịch vụ mà hệ thống phải cung cấp.
- **Các yêu cầu phi chức năng:** các ràng buộc mà hệ thống phải tuân theo.



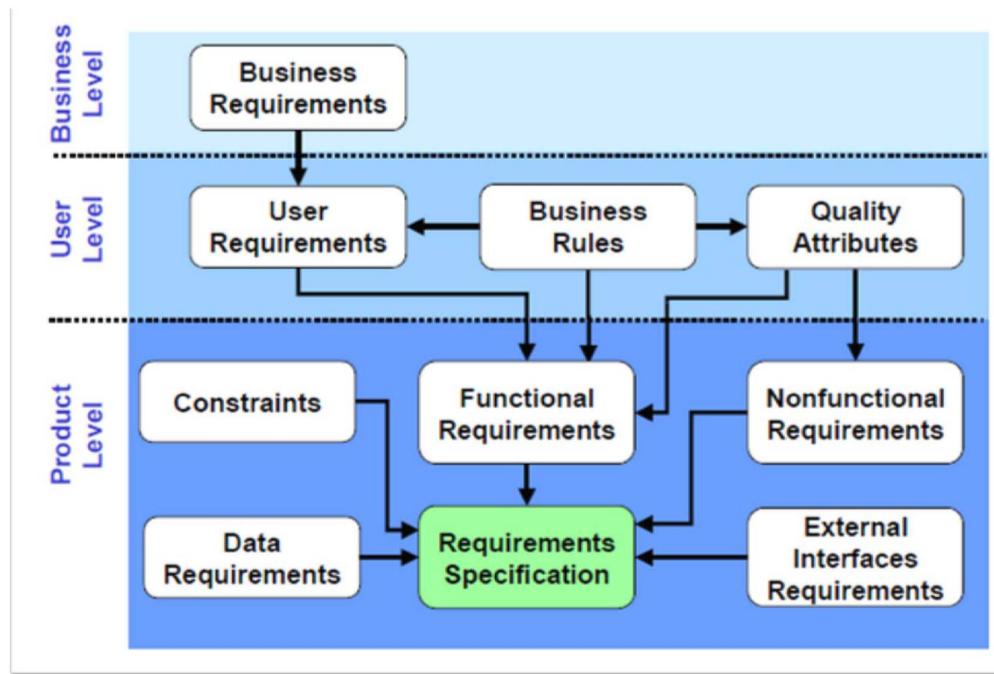
Đánh giá các yêu cầu

Đánh giá các yêu cầu phần mềm liên quan với việc cho biết các yêu cầu thực sự định nghĩa hệ thống đáp ứng đòi hỏi của khách hàng. Nếu việc đánh giá này không chính xác, các lỗi trong phần yêu cầu sẽ truyền tới thiết kế hệ thống và triển khai hệ thống. Chi phí sửa chữa lỗi sẽ rất lớn. Sự thay đổi về yêu cầu ngụ ý rằng việc thiết kế và triển khai cũng phải thay đổi theo. Một số khía cạnh của yêu cầu cần phải kiểm chứng:

- **Giá trị:** người dùng có thể nghĩ rằng hệ thống cần một số chức năng, tuy nhiên sau một số phân tích, có thể xác định các chức năng khác cần được đưa vào. Do hệ thống có nhiều loại người sử dụng nên có các yêu cầu khác nhau và không thể tránh khỏi sự thỏa hiệp các nhu cầu đó.
- **Chắc chắn:** mọi yêu cầu không được mâu thuẫn với các yêu cầu khác
- **Hoàn chỉnh:** định nghĩa cần phải bao gồm mọi chức năng và các ràng buộc
- **Hiện thực:** không có các yêu cầu đặc biệt đến mức phi hiện thực. Có thể dự đoán trước các phát triển phần cứng, tuy nhiên phát triển phần mềm thì khó dự đoán hơn.
- **Mẫu:** là một mô hình chạy được của hệ thống được trình bày với người sử dụng. Đây là một kỹ thuật đánh giá yêu cầu hiệu quả. Nó cho phép người dùng thử nghiệm với hệ thống. Việc đánh giá lại yêu cầu không nên được coi là công việc tiếp theo của tư liệu hóa yêu cầu sau khi đã hoàn thành. Các xem xét về yêu cầu định kỳ liên quan với người dùng và kỹ sư phần mềm luôn cần thiết.

3.2 KỸ THUẬT LẤY YÊU CẦU

Khi lấy yêu cầu, thường ta để ý đến các mức yêu cầu:



3.2.1 PHỎNG VĂN

Phỏng vấn là việc tập hợp một số lượng ít người - thường với một hoặc hai người - cho một thời gian cố định với một mục đích cụ thể. Trong quá trình phỏng vấn, các câu hỏi có thể được thay đổi. Bạn có thể đánh giá được cảm nhận của họ, động cơ và thói quen đối với các phòng ban, quá trình quản lý, ứng dụng hoặc các thực thể khác đáng chú ý. Kiểu phỏng vấn được xác định bởi kiểu thông tin yêu cầu. Phỏng vấn nên được dàn dắt sao cho cả hai bên tham gia đều cảm thấy thỏa mãn với kết quả. Được chuẩn bị kỹ đồng nghĩa là hiểu rõ người đang được phỏng vấn, do đó bạn không làm họ bối rối và bạn có thể có vài câu hỏi ban đầu được chuẩn bị cho dù không phải là tất cả.

3.2.2 HỌP NHÓM

Họp nhóm là việc tập hợp ba hoặc nhiều hơn một số người cho một thời hạn nhất định để thảo luận một số chủ đề. Họp nhóm có thể vừa bổ sung và thay thế phỏng vấn. Nó bổ sung phỏng vấn bằng cách cho phép nhóm kiểm tra lại các kết quả phỏng vấn cá nhân. Nó có thể thay thế cho phỏng vấn

bằng cách cung cấp một diễn đàn cho các người sử dụng cùng tìm ra các đòi hỏi và các giải pháp cho ứng dụng.

3.2.3 QUAN SÁT

Quan sát là hoạt động có thể tiến hành thủ công hoặc tự động như sau:

- Theo cách thủ công, người quan sát ngồi tại một chỗ và ghi chép các hoạt động, các bước xử lý công việc. Ghi chép hoặc bǎng ghi hình được dùng để phân tích cho các sự kiện, các mô tả động từ chính, hoặc các hoạt động chỉ rõ lý do, công việc hoặc các thông tin về công việc.
- Theo cách tự động, máy tính sẽ lưu giữ các chương trình thường trú, lưu lại vết của các chương trình được sử dụng, email và các hoạt động khác được xử lý bởi máy. Các file nhật ký của máy sẽ được phân tích để mô tả công việc.

3.2.4 CÔNG VIỆC TẠM THỜI

Với một công việc tạm thời, ta có được nhận thức đầy đủ hơn về các nhiệm vụ, đồng thời nó cung cấp cho ta kinh nghiệm thực tế. Đầu tiên, ta học các thuật ngữ và ngữ cảnh sử dụng nó. Thời gian tiến hành thường từ 2 tuần đến 1 tháng đủ dài để bạn có thể quen thuộc với phần lớn các công việc thông thường và các tình huống ngoại lệ nhưng không được quá dài để trở thành chuyên gia thực sự đối với công việc

3.2.5 ĐIỀU TRA QUA CÂU HỎI

Điều tra qua câu hỏi là xây dựng các câu hỏi để phỏng vấn trên giấy hoặc máy tính. Các câu hỏi được dùng để nhận các thông tin từ số lượng lớn người sử dụng và thường ở dạng khả năng lựa chọn, người trả lời chỉ việc đánh dấu. Các mục câu hỏi, như là phỏng vấn, có thể là câu hỏi mở hoặc câu hỏi đóng nhưng không chỉ rõ tên, dẫn đến các câu trả lời trung thực hơn nhiều phỏng vấn.

3.2.6 XEM XÉT TÀI LIỆU

Khái niệm tài liệu ám chỉ tới các cẩm nang, quy định, các thao tác chuẩn mà tổ chức cung cấp như là hướng dẫn cho các nhà quản lý và nhân viên.

Các tài liệu thực sự hữu ích cho các kỹ sư phần mềm để học về các lĩnh vực mà họ chưa từng có kinh nghiệm. Nó có thể hữu ích cho việc xác định các câu hỏi về quá trình thao tác và sản xuất. Tài liệu đưa ra các thông tin khách quan.

3.2.7 XEM XÉT PHẦN MỀM

Khi các ứng dụng cũ phải được thay thế các phần mềm mới, việc nghiên cứu các phần mềm đã tồn tại cung cấp cho chúng ta các thông tin về quá trình xử lý công việc hiện thời và các mở rộng có ràng buộc bởi thiết kế phần mềm.

3.3 KẾT QUẢ CỦA GIAI ĐOẠN LẤY YÊU CẦU

3.3.1 KẾT QUẢ

Khi lấy yêu cầu, kết quả của giai đoạn này bao gồm:

- Danh sách các actor
- Danh sách các use case
- Use case diagram
- Cho từng Use Case:
 - Phát thảo giao diện
 - Vẽ sơ đồ hoạt động
 - Viết đặc tả Use case

3.3.2 ĐẶC TẨU UC

Đặc tả một Use Case thường được thực hiện trong văn bản. Đây là lời đặc tả đơn giản và nhất quán về việc các actor và các Use Case (hệ thống) tương tác với nhau ra sao. Nó tập trung vào ứng xử đối ngoại của hệ thống và

không đề cập tới việc thực hiện nội bộ bên trong hệ thống. Ngôn ngữ và các thuật ngữ được sử dụng trong lời miêu tả chính là ngôn ngữ và các thuật ngữ được sử dụng bởi khách hàng/người dùng.

Nội dung đặc tả UC thường bao gồm:

- **Mục đích của Use Case:** Mục đích chung cuộc của Use Case là gì? Cái gì cần phải được đạt tới? Use Case nói chung đều mang tính hướng mục đích và mục đích của mỗi Use Case cần phải rõ ràng.
- **Use Case được khởi chạy như thế nào:** Tác nhân nào gây ra sự thực hiện Use Case này? Trong hoàn cảnh nào?
- **Chuỗi các thông điệp giữa actor và Use Case:** Use Case và các actor trao đổi thông điệp hay sự kiện nào để thông báo lẫn cho nhau, cập nhật hoặc nhận thông tin và giúp đỡ nhau quyết định? Yếu tố nào sẽ miêu tả dòng chảy chính của các thông điệp giữa hệ thống và actor, và những thực thể nào trong hệ thống được sử dụng hoặc bị thay đổi?
- **Dòng chảy thay thế trong một Use Case:** Một Use Case có thể có những dòng thực thi thay thế tùy thuộc vào điều kiện. Hãy nhắc đến các yếu tố này, nhưng chú ý đừng miêu tả chúng quá chi tiết đến mức độ chúng có thể "che khuất" dòng chảy chính của các hoạt động trong trường hợp cẩn thận. Những động tác xử lý lỗi đặc biệt sẽ được miêu tả thành các Use Case khác.

- **Use Case sẽ kết thúc với một giá trị đối với tác nhân như thế nào:** Hãy miêu tả khi nào Use Case được coi là đã kết thúc, và loại giá trị mà nó cung cấp đến tác nhân.

Ví dụ:

Chi tiết tài khoản: // tên Use Case

Số Use Case: UCSEC35

Miêu tả ngắn: // miêu tả ngắn gọn Use Case

Use Case "chi tiết tài khoản" cho phép nhân viên nhà băng xem các chi tiết của một tài khoản mà anh ta định tìm hiểu.

Dòng chảy các sự kiện: // dòng logic chung

Nhân viên chọn Chi Tiết Tài Khoản trên menu. Một con đường khác để chỉ ra các thông tin chi tiết của tài khoản là gọi từ Màn Hình Tóm Tắt Thông Tin Về Tài Khoản (xem Use Case số UCSEC99).

Dòng hành động chính: // dòng logic chi tiết.

Mỗi khách hàng sẽ có một số định danh gọi là CustomerId. Một khách hàng có thể có nhiều tài khoản. Sau khi nhân viên nhập CustomerId vào hệ thống, màn hình phải in ra tất cả những tài khoản thuộc về khách hàng này và thuộc về nhà băng ABC, rải rác tại tất cả các chi nhánh. Khi chọn tiếp loại tài khoản và số tài khoản, các chi tiết của tài khoản mong muốn sẽ được in ra.

Loại tài khoản tiết kiệm: Nếu loại tài khoản được chọn là tài khoản tiết kiệm, thì theo Use Case số UCSEC45, các chi tiết sau đây sẽ được in ra:

- Mức tiền hiện có
- Các tờ sec chưa thanh toán
- Lượng tiền tín dụng được phép
- Lượng tiền lãi cho tới ngày hôm nay
- Lượng tiền tối thiểu cần phải có trong tài khoản

Loại tài khoản đầu tư: Nếu loại tài khoản được chọn là loại đầu tư, thì theo Use Case số UCSEC46, các chi tiết sau đây sẽ được in ra.

- Hạn đầu tư
- Số tiền đầu tư
- Ngày đầu tư
- Lượng tiền cuối hạn
- Ngày cuối hạn
- Tỷ lệ lời

Dòng hành động thay thế: // chuỗi logic thay thế

Không tìm thấy chi tiết: Khi chọn một số tài khoản không thích hợp (không có tài khoản tương ứng) dù vì lý do chức năng hay kỹ thuật, theo Use Case số UCSEC12, hệ thống sẽ đưa ra một màn hình báo lỗi.

Điều kiện thoát: // Use Case kết thúc như thế nào?

Nút Thoát: Khi chọn nút thoát, người sử dụng sẽ quay trở lại màn hình chính.

Nút Xem Thêm: Khi chọn nút này, người sử dụng sẽ được yêu cầu chọn loại tài khoản từ một danh sách đồ xuống.

Nút Xem Giao Dịch: Khi chọn nút này, người sử dụng sẽ được chuyển sang màn hình "Giao dịch" và theo Use Case số UCSEC91, màn hình sẽ chỉ ra những giao dịch đã xảy ra đối với tài khoản, bên cạnh những chi tiết chính của tài khoản.

Nút Yêu Cầu In Kết Quả: Khi chọn phần thực đơn này, kết quả giao dịch theo Use Case số UCSEC70 sẽ được in ra ở một máy in địa phương nối trực tiếp với máy tính của nhân viên.

Các yêu cầu đặc biệt: // các yêu cầu đặc biệt

Theo Use Case số UCSEC110, hệ thống có khả năng in lên màn hình bằng những ngôn ngữ khác. Chức năng này sẽ được kích hoạt khi người sử dụng chọn mục Ngoại Ngữ trên menu.

Điều kiện trước đó: // điều xảy ra trước khi Use Case được thực hiện

Bảo an: Người sử dụng (nhân viên tiếp khách) được cung cấp một số định danh riêng biệt để truy nhập vào hệ thống.

Dịch chuyển: Người sử dụng chỉ đến được màn hình Chi Tiết Tài Khoản sau khi đã truy nhập thành công và Identify thành công.

Điều kiện sau đó: // điều gì xảy ra sau khi Use Case được thực hiện?

Hệ thống sẽ không lưu trữ lại bất kỳ một thông tin nào liên quan tới khách hàng lên đĩa cứng cục bộ.

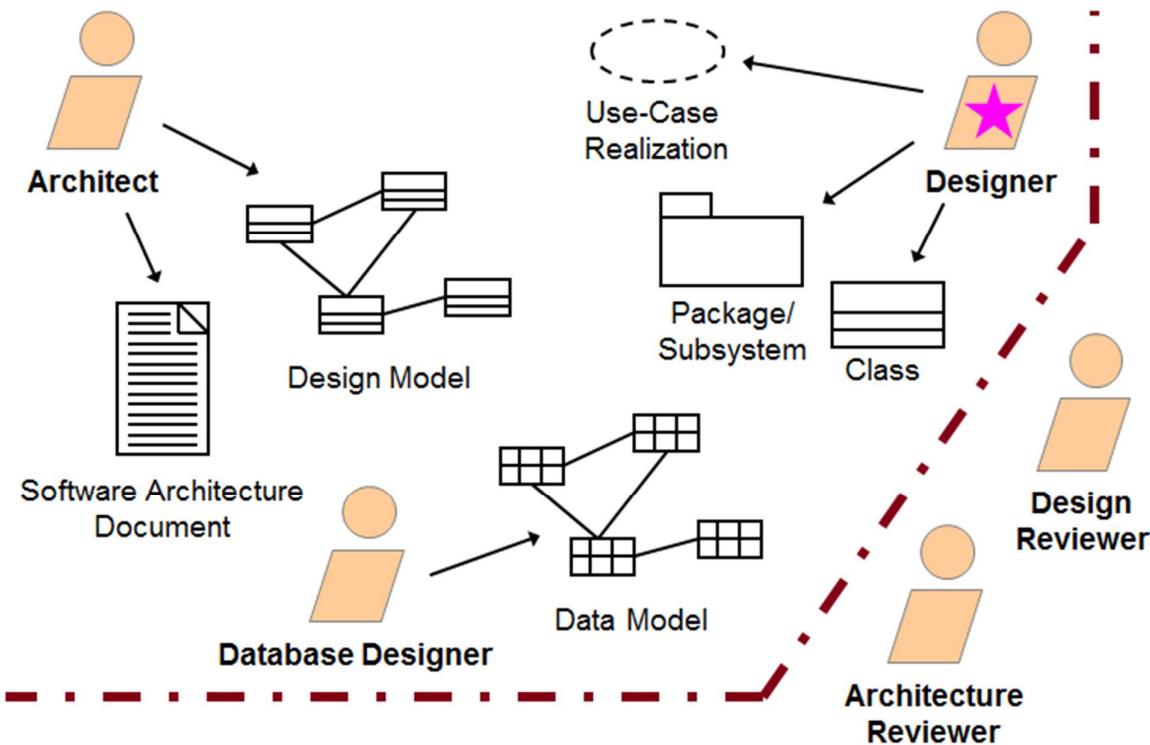
BÀI TẬP ĐỀ NGHỊ

Câu 1: Lấy yêu cầu cho hệ thống quản lý thư viện

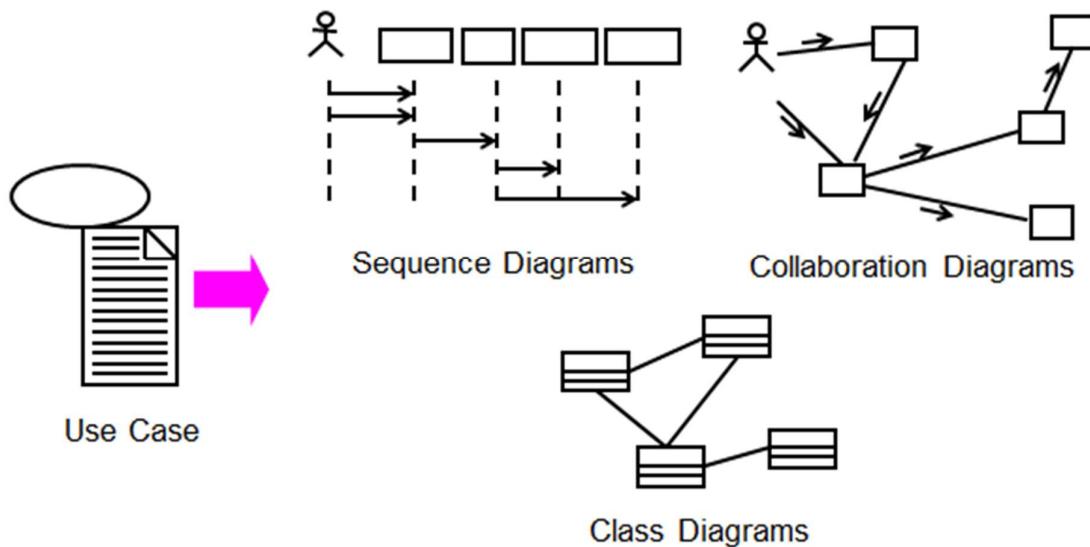
BÀI 4: PHÂN TÍCH & THIẾT KẾ

4.1 GIỚI THIỆU

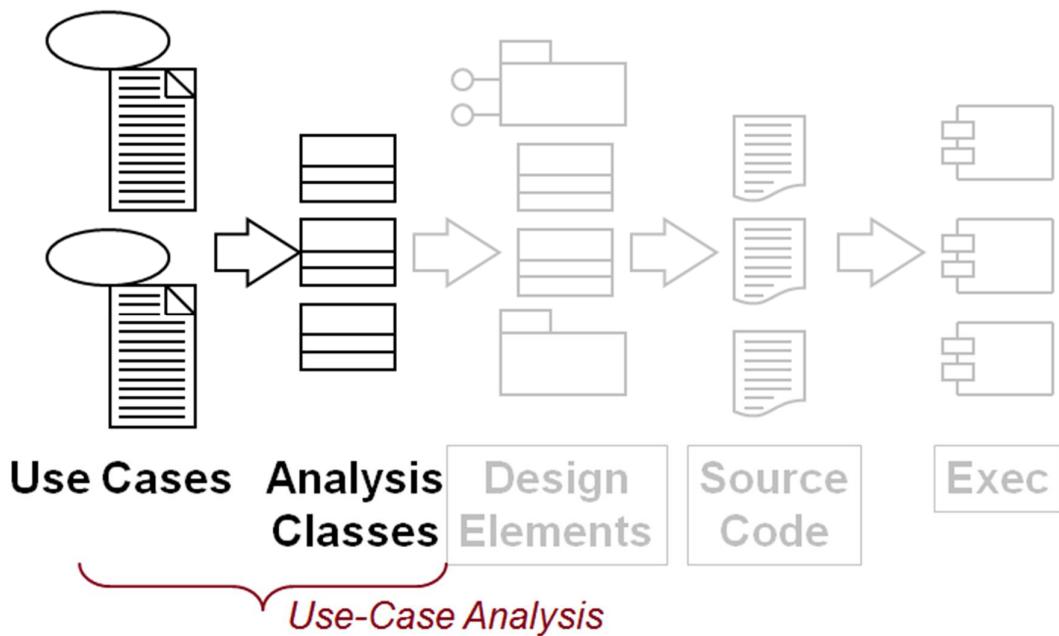
Các công việc và vai trò phải thực hiện trong giai đoạn phân tích thiết kế:



Trong phần này, chúng ta sẽ tìm hiểu phân tích và thiết kế theo định hướng từng Use Case. Cho từng Use Case, chúng ta sẽ phân tích thiết kế ở trạng thái tĩnh và trạng thái động:



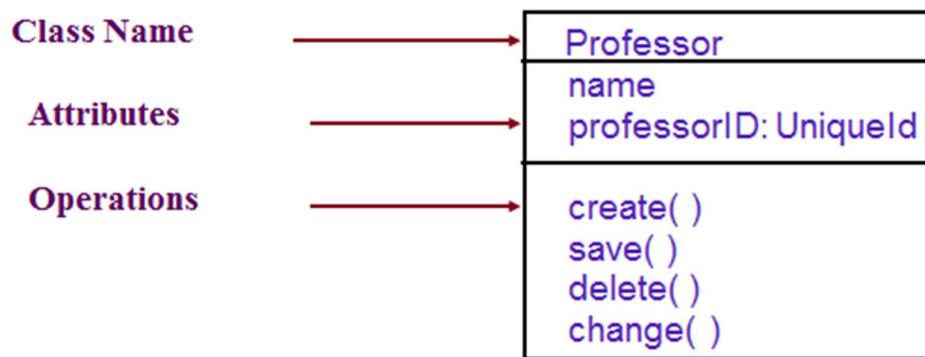
Các bước dùng cho quá trình phân tích thiết kế như sau:



4.2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG Ở TRẠNG THÁI TĨNH

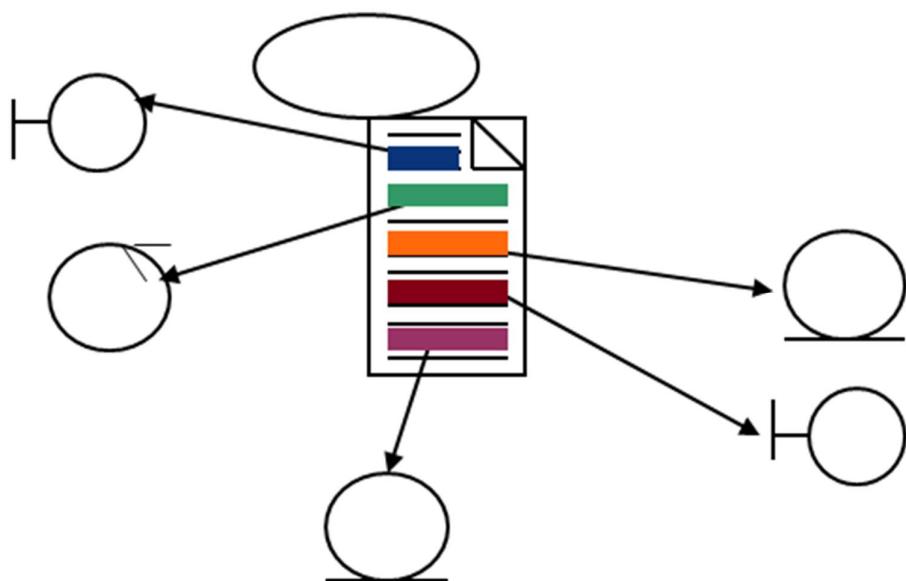
Khi phân tích hệ thống ở trạng thái tĩnh, ta tiến hành tìm các lớp phân tích và vẽ sơ đồ class diagram, tìm kiếm các thuộc tính cho các lớp này.

Lớp bao gồm 3 thành phần như sau:



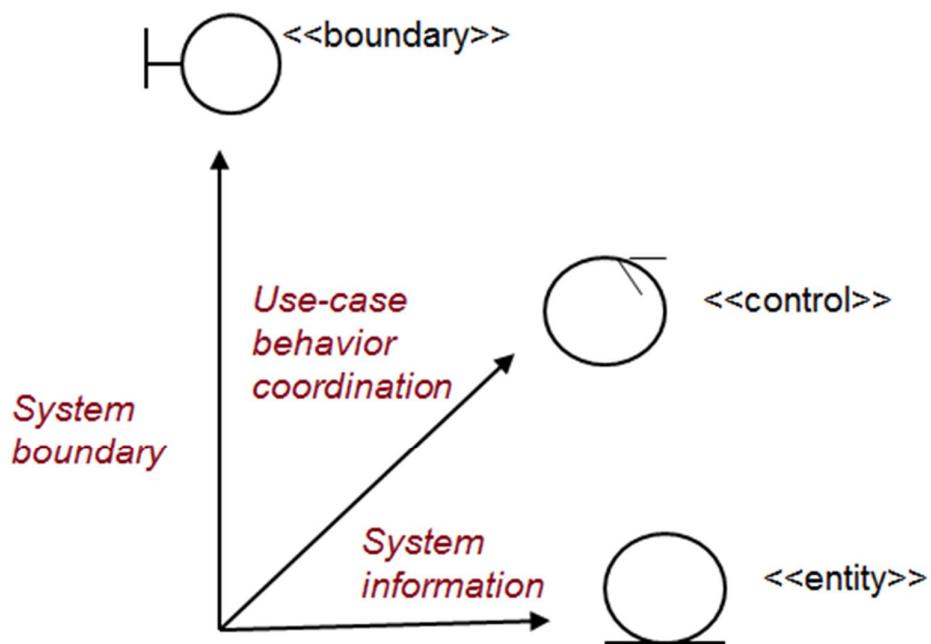
Với mỗi một UC, có thể tồn tại 3 loại lớp: Lớp boundary, lớp control và lớp entity.

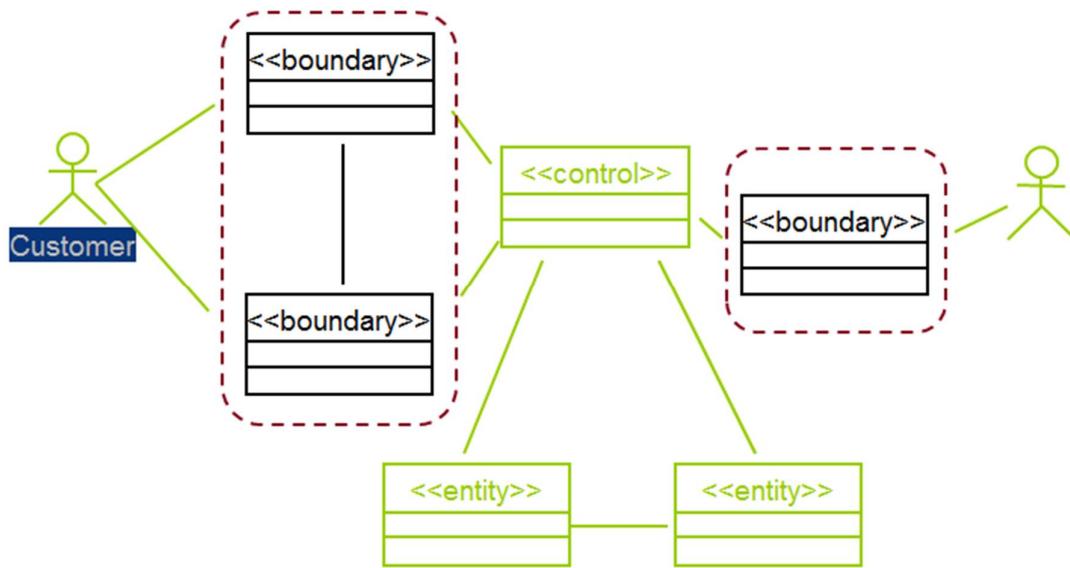
Ký hiệu:



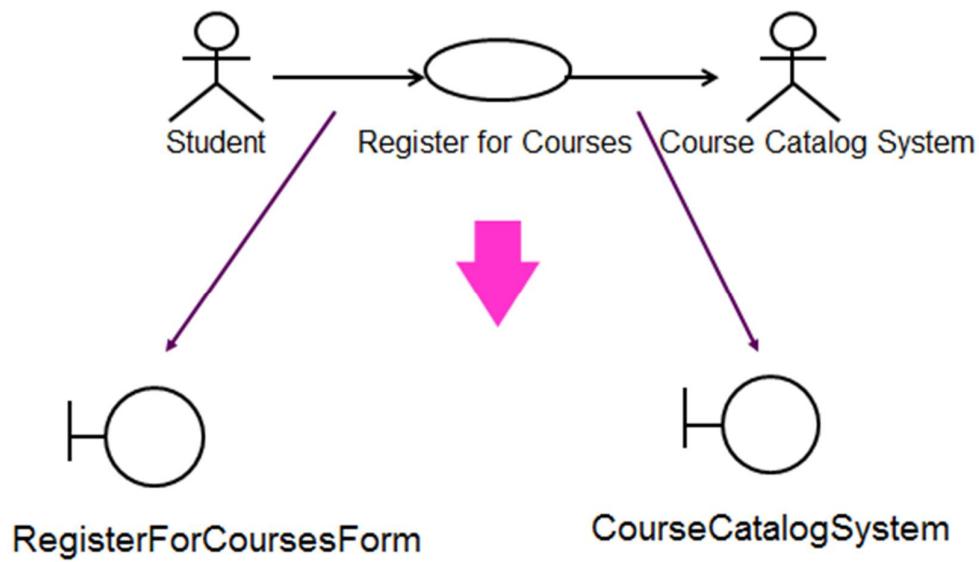
- Lớp biên (boundary): Dùng để mô tả sự tương tác giữa các actor với hệ thống, lớp biên có thể là: Giao diện người dùng, Lớp tương tác với các thiết bị phần cứng, Lớp tương tác với các hệ thống khác.
- Lớp điều khiển (controller): thông thường 1 UC có 1 lớp Controller điều khiển hoạt động của UC
- Lớp Thực thể (Entity): Mô tả các thực thể chứa dữ liệu cần thiết để hiện thực UC.

Các lớp này được thể hiện trong sơ đồ sau:

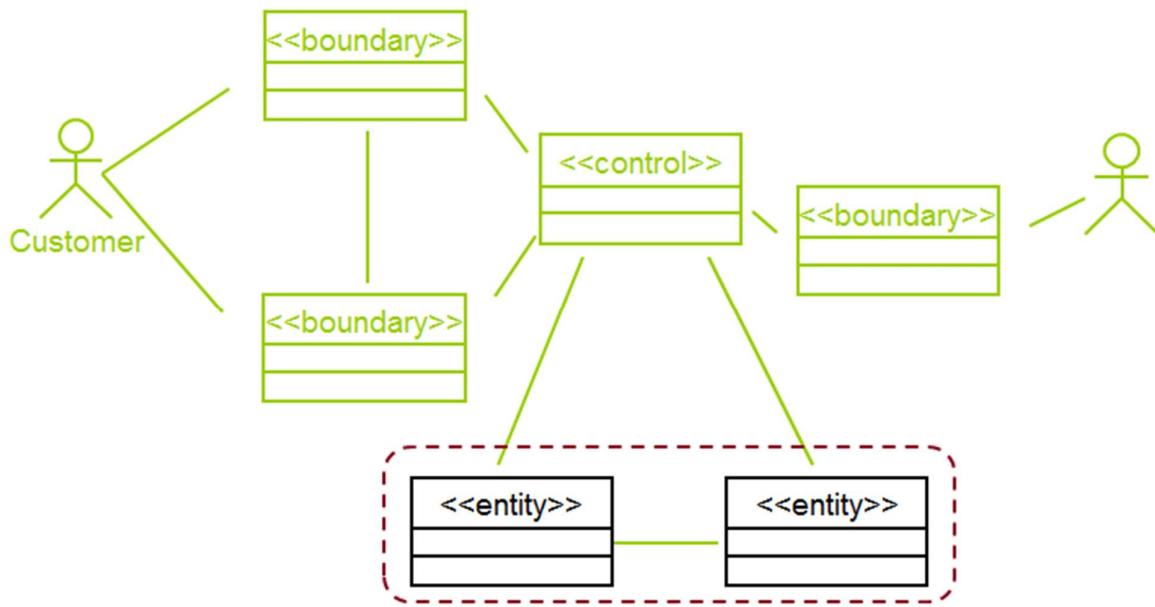




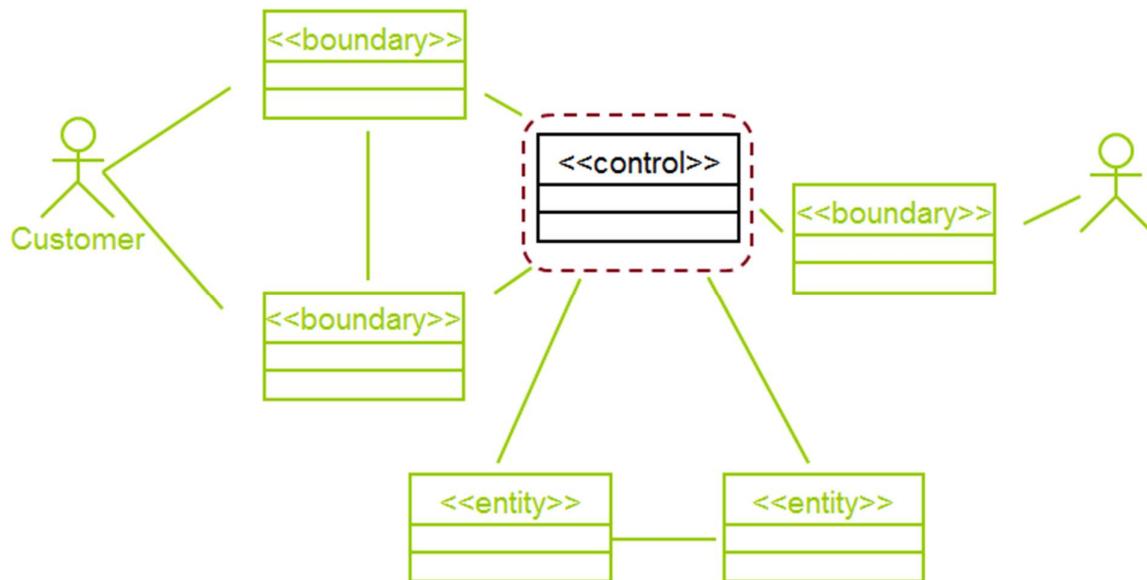
Ví dụ:



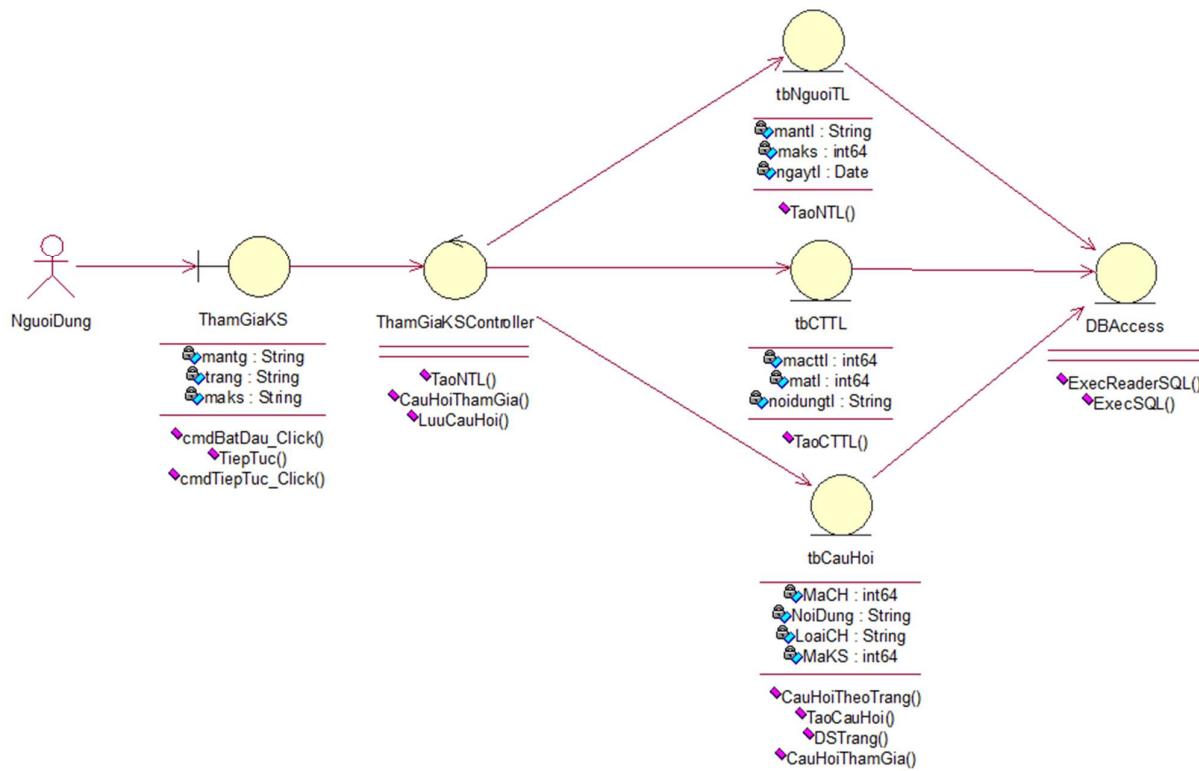
Entity class:



Control class

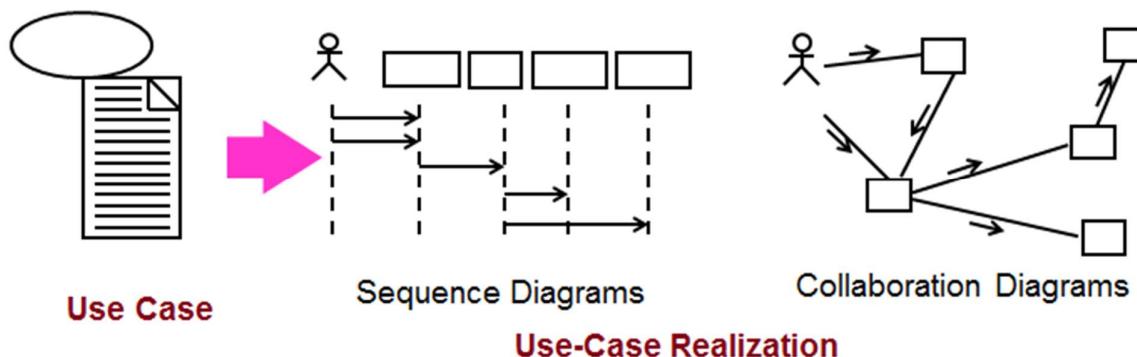


Sau khi phân tích, dựa vào đặc tả ta có thể tìm kiếm các attribute của các lớp. Sơ đồ class diagram được vẽ như sau:



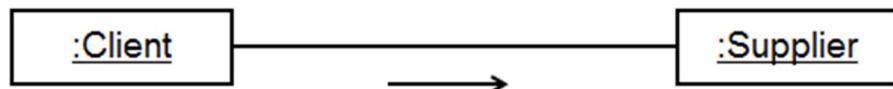
4.3 PHÂN TÍCH THIẾT KẾ HỆ THỐNG Ở TRẠNG THÁI ĐỘNG

Khi phân tích và thiết kế ở trạng thái động, ta vẽ sơ đồ tuần tự & sơ đồ giao tiếp để tìm kiếm các hàm (method) cho các lớp phân tích ở trên.



Mỗi một thông điệp giữa các object là lời gọi hàm để hiện thực UC.

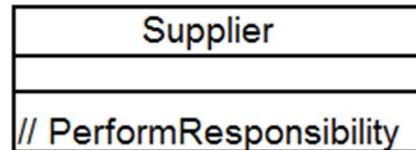
Interaction Diagram



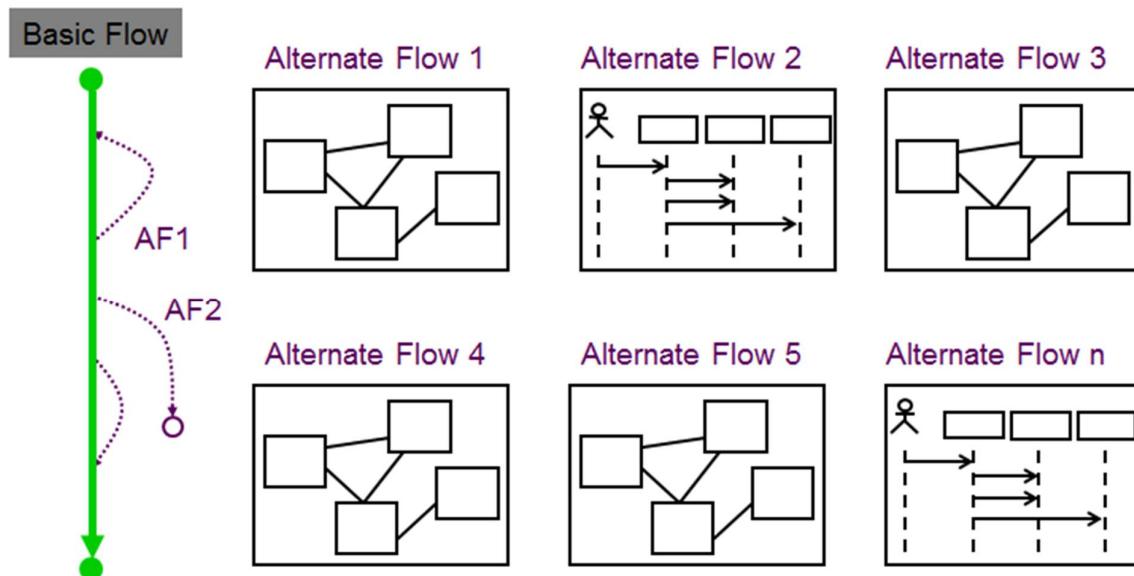
// PerformResponsibility



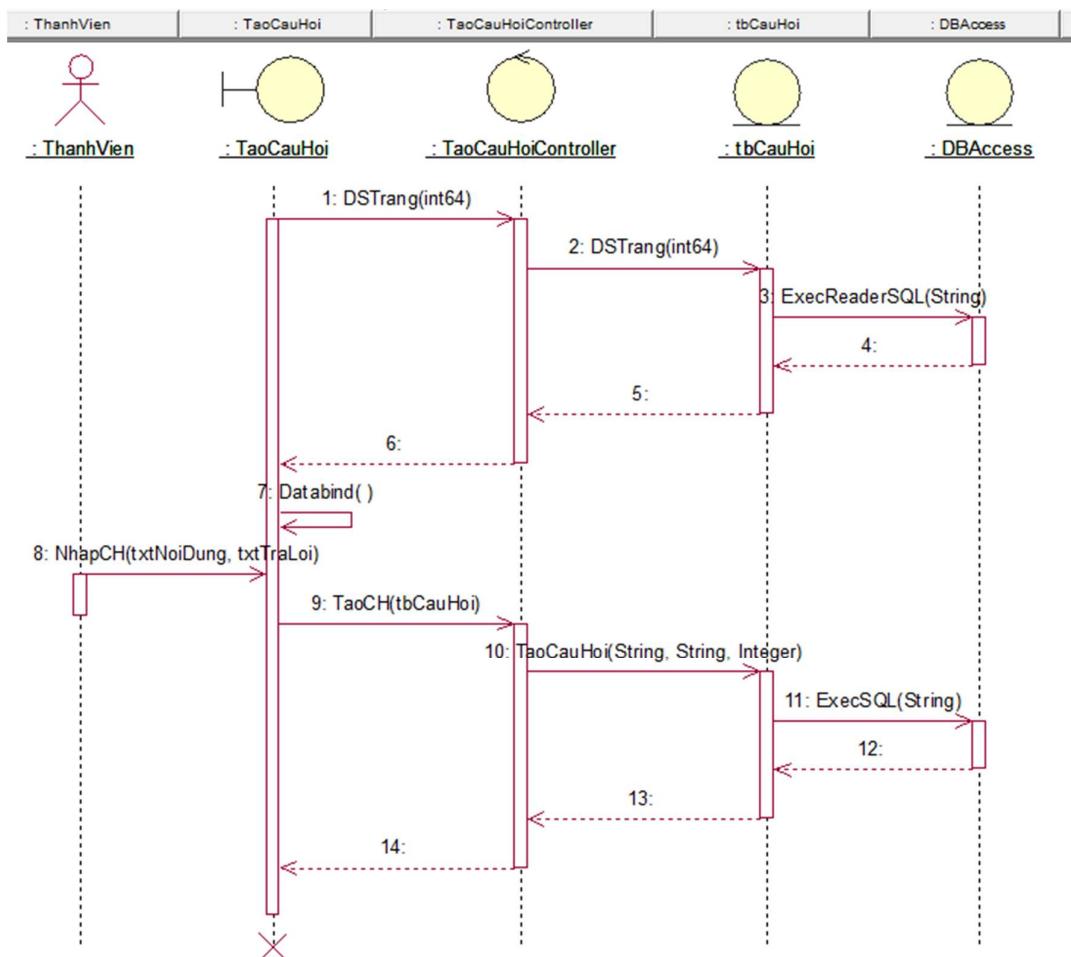
Class Diagram



Nếu UC có nhiều luồng thông tin, ta phải vẽ nhiều sơ đồ:



Ở giai đoạn thiết kế, ta thêm vào các tham số cho hàm, kiểu dữ liệu cho các tham số, kiểu kết quả trả về.



BÀI TẬP ĐỀ NGHỊ

Câu 1: Phân tích & thiết kế hệ thống đã lấy yêu cầu ở bài 3

TÀI LIỆU THAM KHẢO

1. **Pro ASP.NET 4 in CSharp 2010**, Matthew MacDonald, Adam Freeman, and Mario Szpuszta, appress 2010.
2. **ASP.NET 2.0 Everyday Apps For Dummies**, Doug Lowe, Wiley Publishing, 2007

PHỤ LỤC – ĐỒ ÁN THAM KHẢO

Đề bài: phân tích và thiết kế hệ thống tạo khảo sát online

UC Diagram

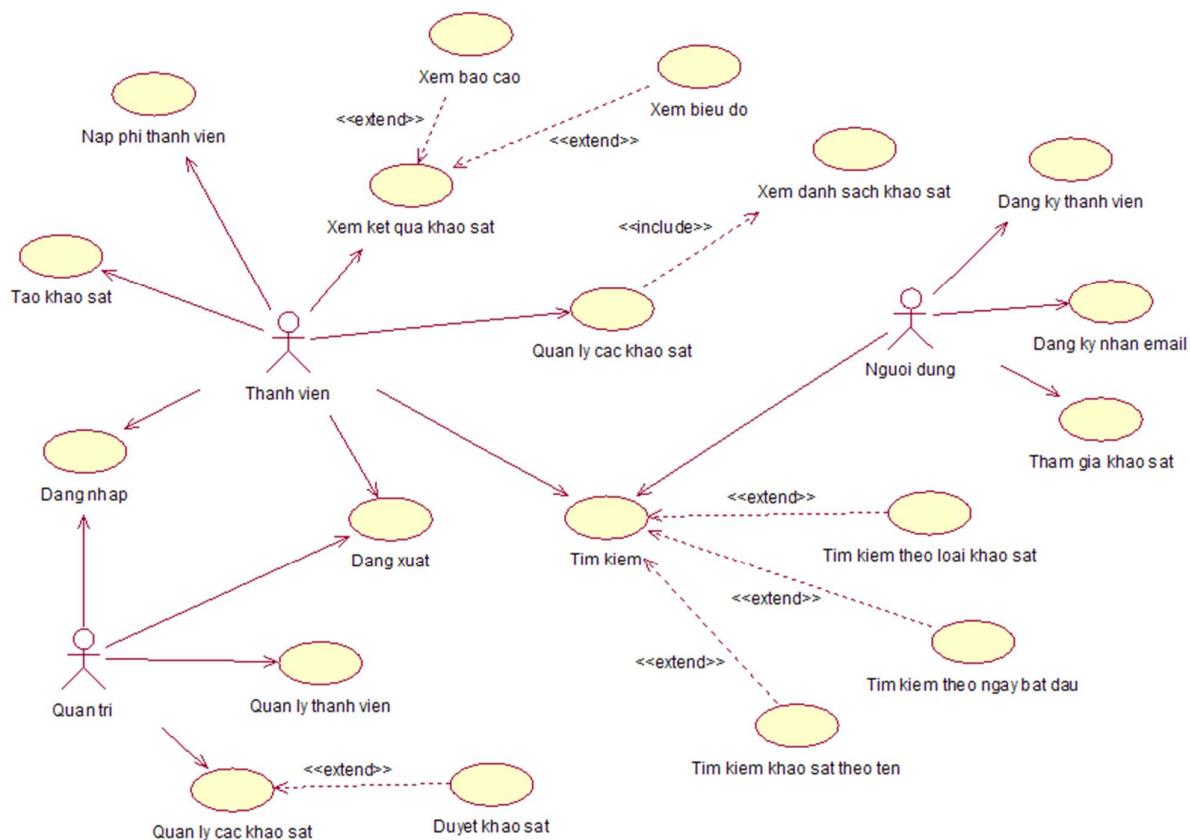
Danh sách Actor:

- Thành viên
- Người dùng
- Quản trị

Danh sách Usecase

- Tạo khảo sát
- Quản lý khảo sát
- Tạo câu hỏi
- Quản lý câu hỏi
- Nạp phí thành viên
- Xem kết quả khảo sát
- Xem biểu đồ
- Tìm kiếm khảo sát
- Quản trị quản lý khảo sát
- Quản trị quản lý thành viên

Sơ đồ Usecase:



Thiết kế chi tiết chức năng:

Giao diện trang chủ:

Đăng nhập

ESurvey

Tiết kiệm hơn - Hiệu quả hơn

KHẢO SÁT TRỰC TUYẾN

- Rất tốt
- Tốt
- Tạm được

TRANG CHỦ | KHẢO SÁT CỦA BẠN | TRỢ GIÚP

ĐĂNG NHẬP

Tài khoản:

Mật khẩu:

Lưu mật khẩu.

Đăng nhập

TÌM KIẾM

Tên khảo sát:

Tìm kiếm

HỆ THỐNG KHẢO SÁT TRỰC TUYẾN

Để dàng và nhanh
chóng với sự hỗ trợ
từ thư viện khảo
sát mẫu

Cho hàng ngàn
người đăng ký
tham gia trả lời

Và phân tích kết
qua với nhiều loại
báo cáo khác
nhau

Khảo sát nhiều người tham gia

- ➡ Khảo sát mới nhất
- ➡ Khảo sát về xu hướng thời trang
trẻ trai lì lợm
- ➡ Khảo sát về xu hướng thời trang
trẻ
- ➡ Khảo sát về tài nguyên mạng
- ➡ khảo sát

Khảo sát mới nhất

- ➡ Khảo sát mới nhất
- ➡ Khảo sát về xu hướng thời trang
trẻ trai lì lợm
- ➡ Khảo sát về xu hướng thời trang
trẻ
- ➡ Khảo sát về tài nguyên mạng
- ➡ khảo sát

Khảo sát đang hoạt động

- ➡ Nghiên cứu thị trường hàng hóa
- ➡ khảo sát miễn phí
- ➡ tạo khảo sát

Tạo khảo sát

- Phát thảo giao diện:

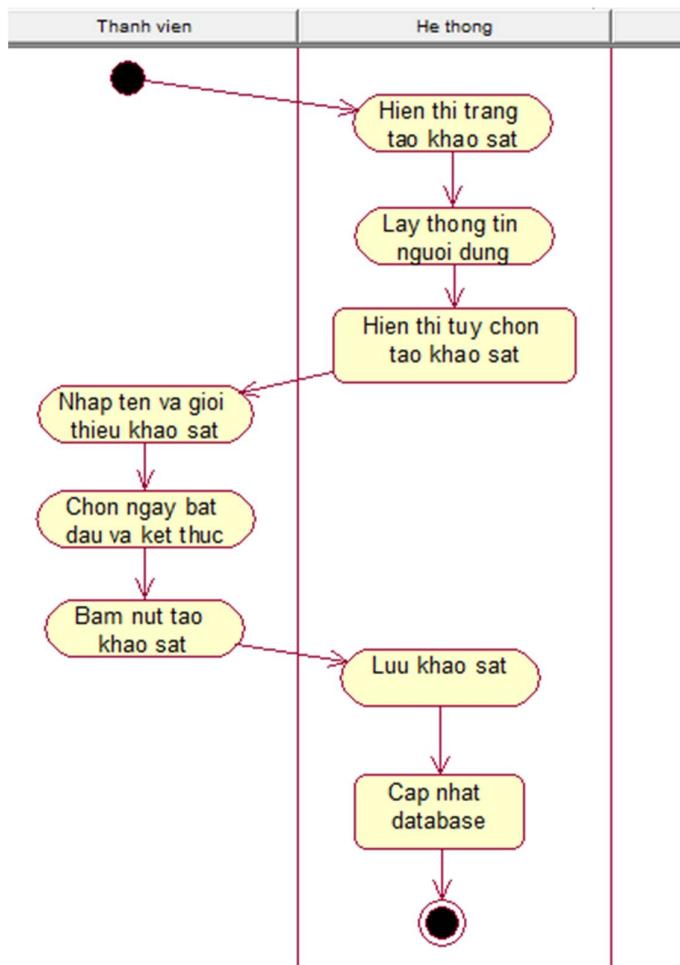
The screenshot displays the ESurvey website's interface for creating a new survey. The main header reads "KHẢO SÁT TRỰC TUYẾN". On the right, there are three checkboxes for rating: "Rất tốt" (Very good), "Tốt" (Good), and "Tạm được" (Temporary). Below the header, there are tabs for "TRANG CHỦ", "QUẢN LÝ KHẢO SÁT", and "TRỢ GIÚP". The "QUẢN LÝ KHẢO SÁT" tab is active. On the left, a sidebar titled "TÀI KHOẢN" shows "Số ngày Vip: 3" and "Số dư tài khoản: 250000", with a "Nạp phí" (Top up) button. Below the sidebar is a search bar with a "Tìm kiếm" (Search) button. The main content area is titled "TAO KHẢO SÁT MỚI" and contains fields for "Tên khảo sát" (Survey name), "Giới thiệu" (Introduction), "Ngày bắt đầu" (Start date), and "Ngày kết thúc" (End date). A "Tạo khảo sát" (Create survey) button is located at the bottom of this form. At the very bottom of the page, there are links for "Trang chủ", "Quy định", "Hướng dẫn", and "Sơ đồ trang", followed by the text "Bản quyền ESurvey 2011".

- Đặc tả usecase:

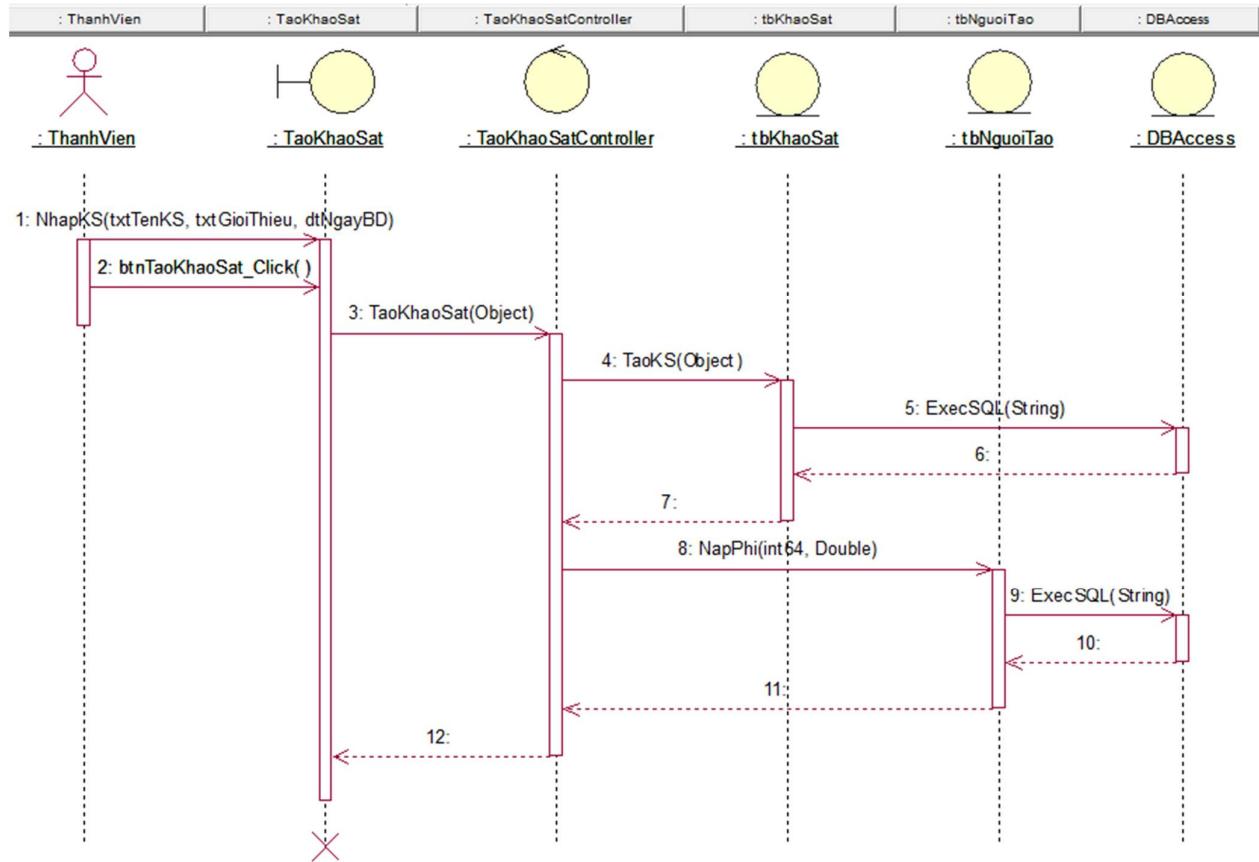
Name	Tạo khảo sát
Description	Thành viên tạo khảo sát
Actor	Thành viên
Pre conditions	<ul style="list-style-type: none"> Đăng nhập vào tài khoản thành viên Hiển thị trang tạo khảo sát
Post conditions	Thông báo kết quả tạo khảo sát
Flow of events	<ol style="list-style-type: none"> Hệ thống lấy thông tin tài khoản người dùng Hệ thống kiểm tra tài khoản. Hệ thống hiển thị tùy chọn tạo khảo sát có tính phí hay không. Người dùng nhập tên khảo sát Người dùng nhập giới thiệu khảo sát Chọn ngày bắt đầu khảo sát Chọn ngày kết thúc khảo sát

	<p>8. Nhấn nút tạo khảo sát</p> <p>8.1 Hệ thống lưu nội dung khảo sát xuống cơ sở dữ liệu</p> <p>8.2 Không lưu được: A1</p> <p>8.3 Thông báo kết quả lên trang web</p>
Alternative flow	<p>A1: Thông báo không tạo được khảo sát. Trở lại trang tạo khảo sát</p> <p>A2: Thông báo không lưu được khảo sát Trở lại trang tạo khảo sát</p>

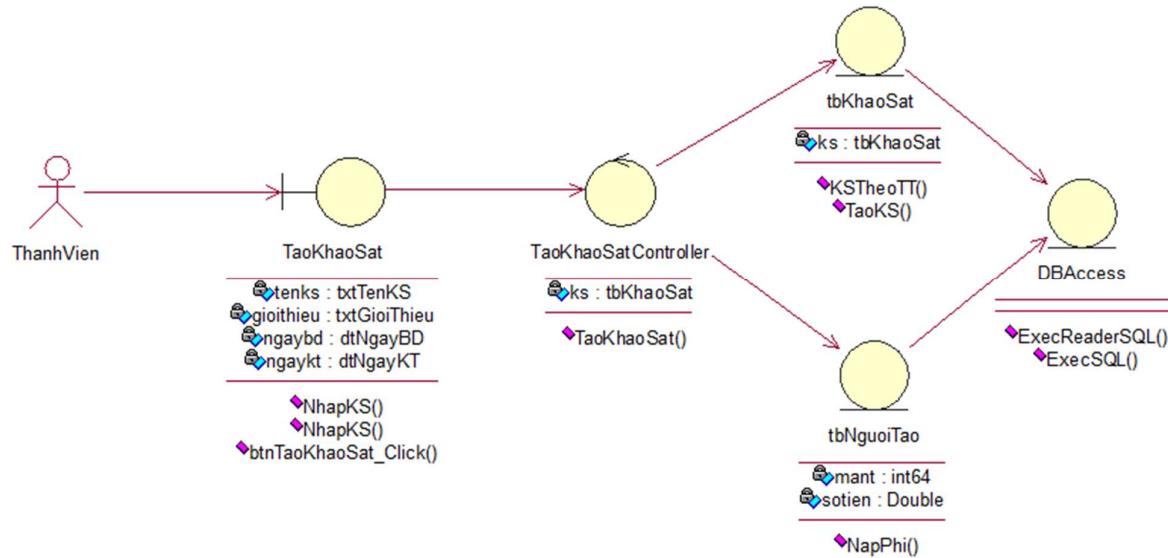
- Activity diagram:



- Sequence diagram:



- Class diagram:



Quản lý khảo sát

Giao diện:

The screenshot shows the ESurvey website interface. At the top right, there is a sign-in button labeled "Đăng nhập". Below it, a blue bar features the ESurvey logo and the text "Tiết kiệm hơn - Hiệu quả hơn". To the right of the logo, the title "KHẢO SÁT TRỰC TUYẾN" is displayed. A feedback scale with three options ("Rất tốt", "Tốt", "Tạm được") is also visible.

The main content area has a header "DANH MỤC" with links to "TRANG CHỦ", "KHẢO SÁT CỦA BẠN", and "TRỢ GIÚP". On the left, there is a sidebar with sections for "Danh mục" (listing "Đang thiết kế", "Chờ duyệt", "Không duyệt", "Đang chạy", and "Đã kết thúc"), "TÀI KHOẢN" (showing "Số ngày Vip: 3" and "Số dư tài khoản: 250000"), and a "Nạp phí" button. Below this is a "TÌM KIẾM" section with a search input field and a "Tim kiếm" button.

The central part of the page displays a table titled "DANH SÁCH KHẢO SÁT" listing six surveys. Each row includes columns for the survey name, creation date, type, and edit/delete options. The surveys listed are:

Tên khảo sát	Ngày Tạo	Loại	Tùy chọn
Khảo sát hàng điện máy	10/08/2011	mễn phí	Sửa Xóa
Khảo sát về nước giải khát THP	12/08/2011	Loại ks	Sửa Xóa
Khảo sát mới về nhu cầu mua hàng qua mạng	12/08/2011	Loại ks	Sửa Xóa
Nghiên cứu thị trường hàng hóa	12/08/2011	Loại ks	Sửa Xóa
Khảo sát về nước hoa	12/08/2011	vip	Sửa Xóa
khảo sát nhân viên văn phòng	12/08/2011	vip	Sửa Xóa

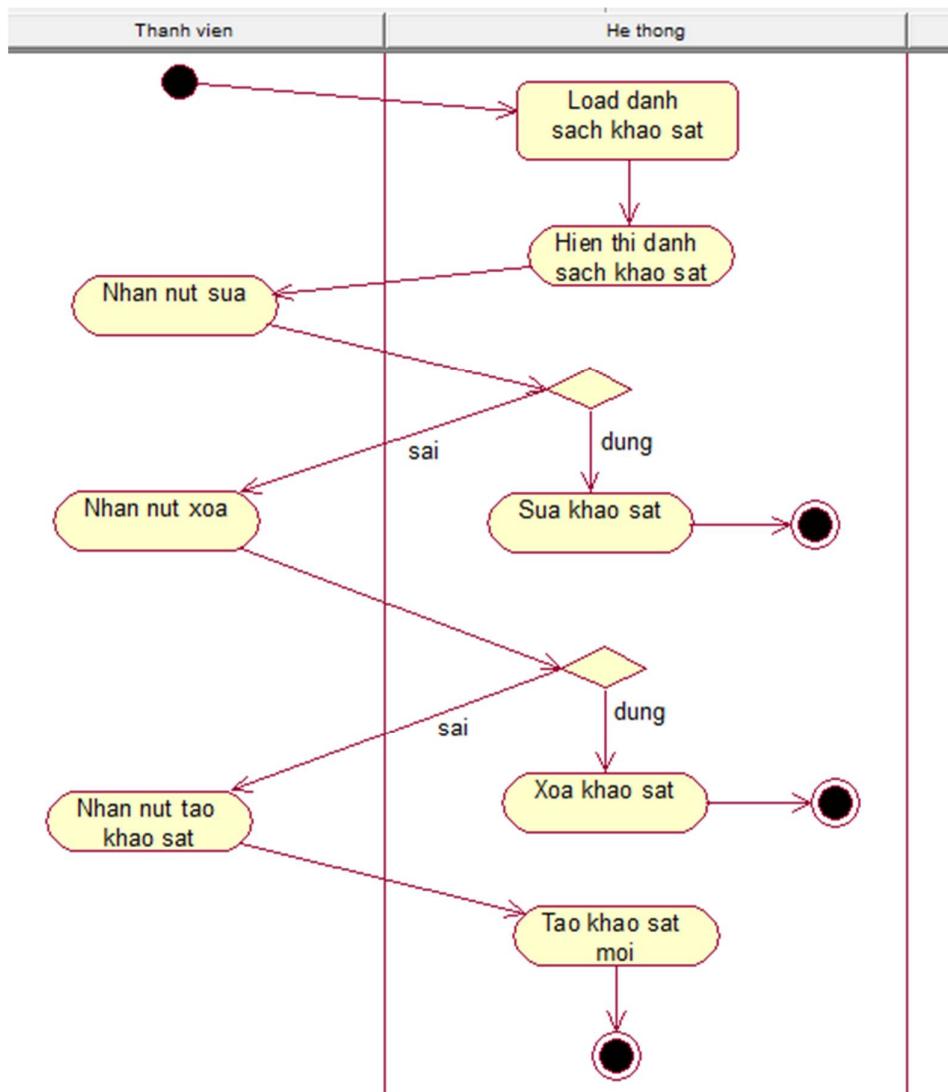
Below the table, there is a page navigation bar showing "Page 1 of 1 (6 items)" and a "Tạo khảo sát" button. At the bottom of the page, there are links to "Trang chủ", "Quy định", "Hướng dẫn", "Sơ đồ trang", and "Bản quyền ESurvey 2011".

- Đặc tả usecase:

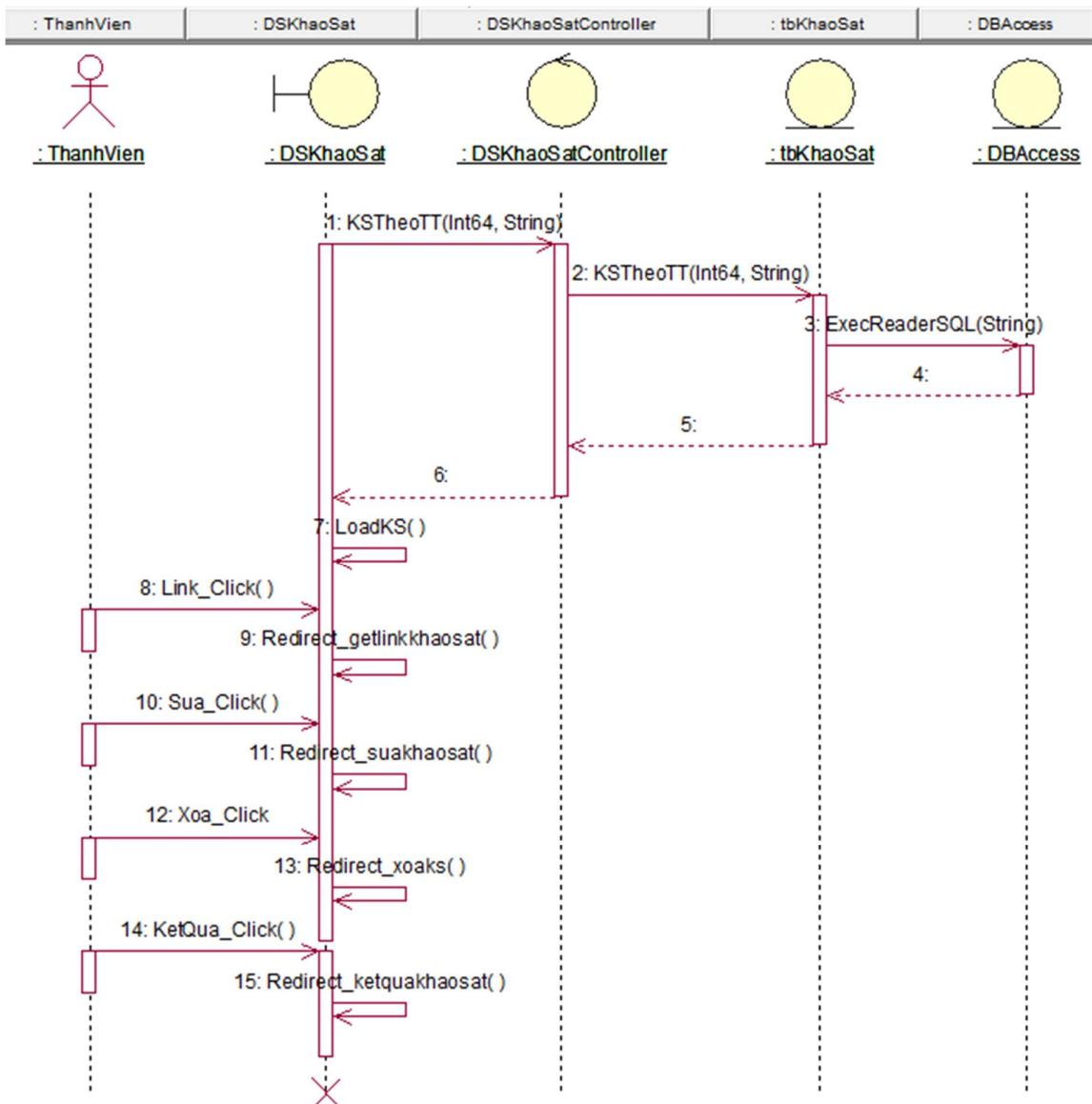
Name	Thành viên quản lý khảo sát
Description	Thành viên quản lý các khảo sát đã tạo
Actor	Thành viên
Pre conditions	<ul style="list-style-type: none"> Đăng nhập vào tài khoản thành viên Hiển thị trang thành viên quản lý khảo sát
Post conditions	Thông báo kết quả thao tác
Flow of events	<ol style="list-style-type: none"> Hệ thống load danh sách khảo sát đã tạo của thành viên. Hệ thống hiển thị danh sách khảo sát đã tạo Người dùng chọn khảo sát Nhấn nút chỉnh sửa Không sửa được: A1 Hệ thống hiển thị trang cập nhật khảo sát Người dùng nhấn nút xoá khảo sát

	9. Hệ thống xoá khảo sát khỏi cơ sở dữ liệu 10. Thông báo kết quả xoá 11. Quay lại trang quản lý khảo sát
Alternative flows	A1: Thông báo không cập nhật được khảo sát A2: Thông báo không xoá được khảo sát

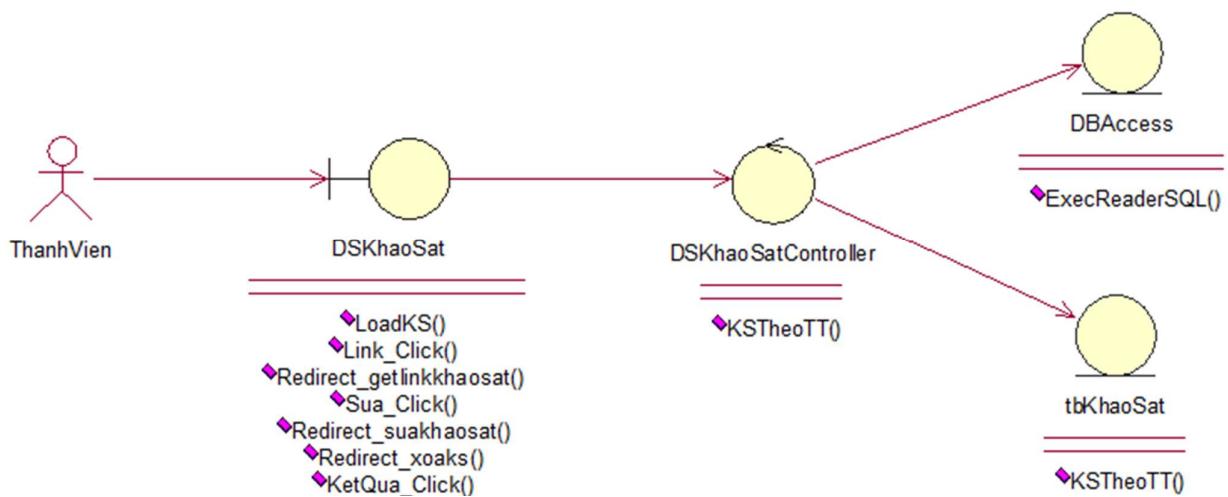
- Activity diagram:



- Sequence diagram:



- Class diagram:



Tạo câu hỏi

- Giao diện

The screenshot shows the ESurvey software interface. At the top, there's a blue header with the logo 'ESurvey' and the tagline 'Tiết kiệm hơn - Hiệu quả hơn'. To the right of the header, there's a section titled 'KHẢO SÁT TRỰC TUYẾN' with three checkboxes: 'Rất tốt', 'Tốt', and 'Tạm được'. Below the header, there are navigation links: 'TRANG CHỦ | QUẢN LÝ KHẢO SÁT | TRỢ GIÚP'. The main content area is titled 'TAO CÂU HỎI' and contains the following fields:

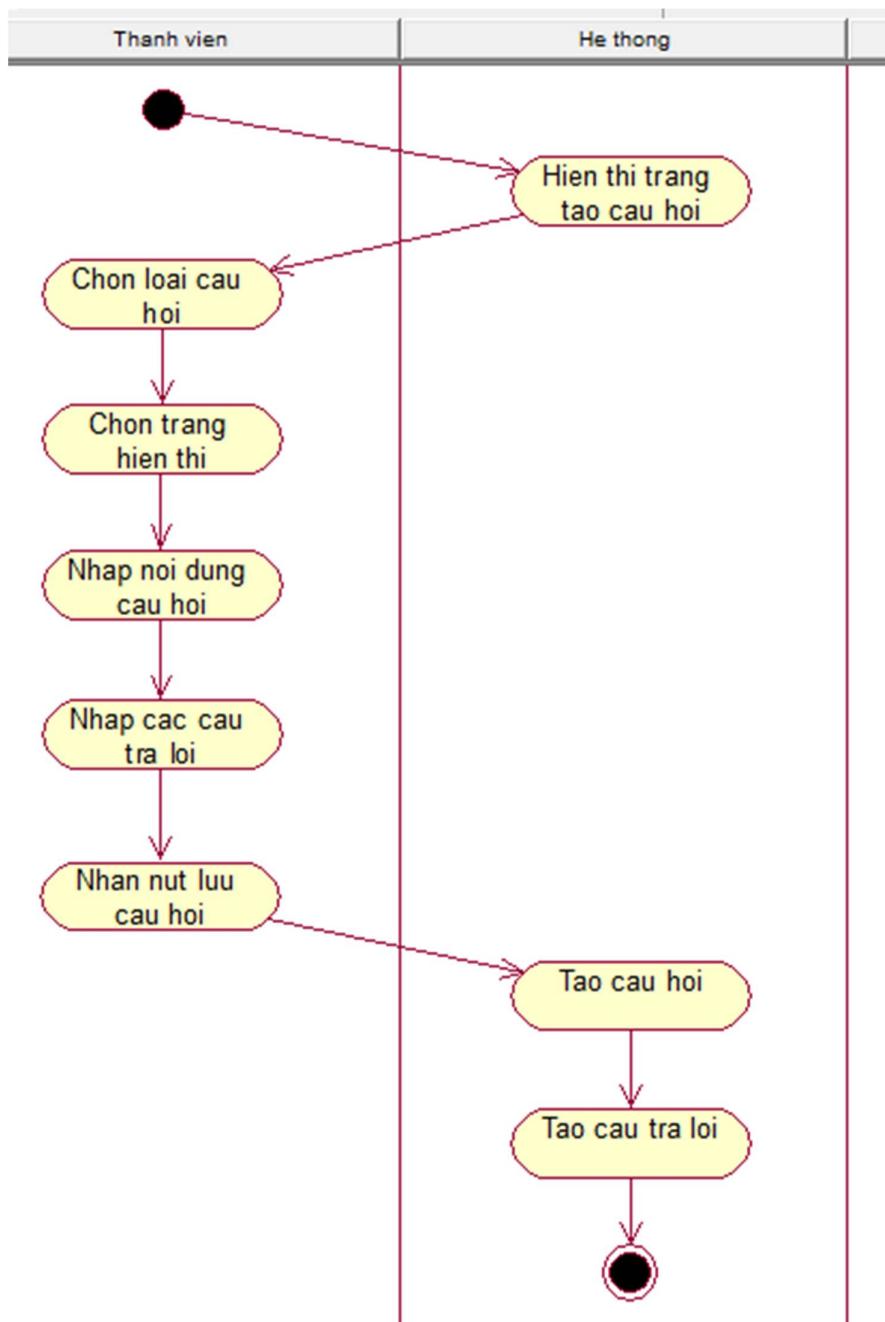
- Khảo sát về nước giải khát THP**
- Loại câu hỏi**: A dropdown menu set to 'Câu hỏi chỉ chọn một trả lời'.
- Trang hiển thị**: A dropdown menu set to '1'. There's also a checkbox for 'Thêm trang 4'.
- Nội dung câu hỏi**: A large text area for entering the question content.
- Các câu trả lời**: A large text area for entering the answer options.
- Tạo câu hỏi**: A blue button at the bottom of the form.

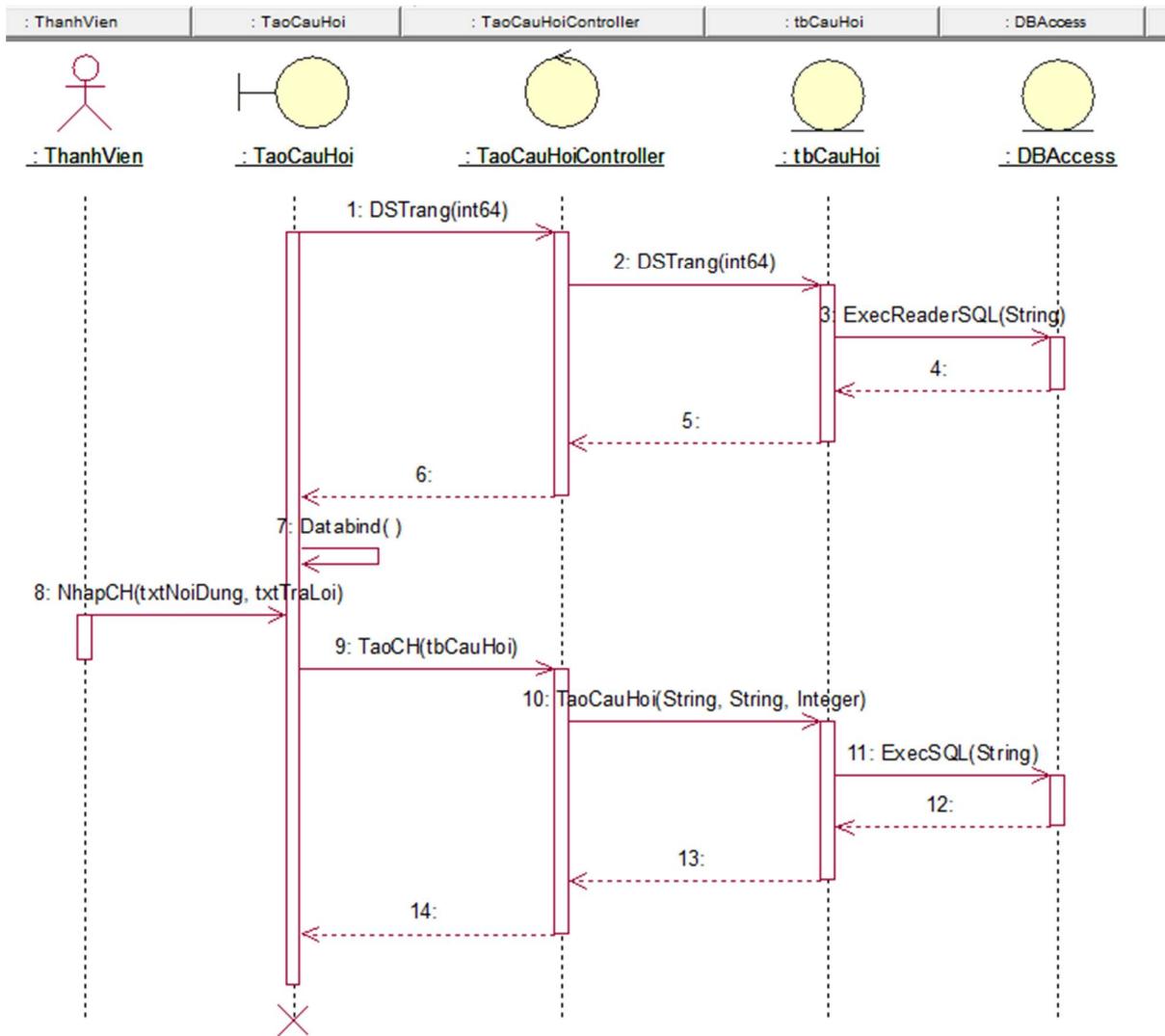
- Đặc tả usecase:

Name	Tạo câu hỏi
Description	Thêm mới câu hỏi khảo sát
Actor	Thành viên
Pre conditions	<ul style="list-style-type: none"> Đăng nhập vào tài khoản thành viên Hiển thị trang danh sách câu hỏi
Post conditions	Thông báo kết quả tạo câu hỏi
Flow of events	<ol style="list-style-type: none"> Hệ thống hiển thị trang tạo câu hỏi Người dùng chọn loại câu hỏi Người dùng chọn trang hiển thị Nhập nội dung câu hỏi

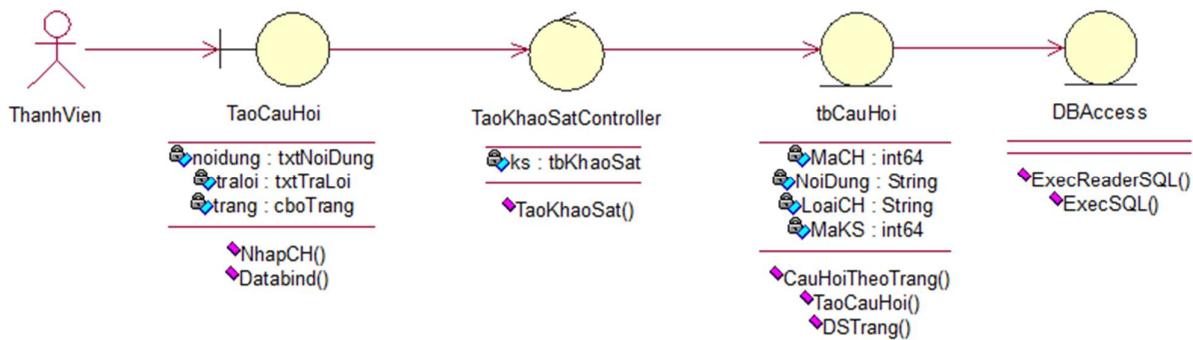
	<p>6. Nhấn nút tạo câu hỏi</p> <p>6.1 Không lưu được câu hỏi : A1</p> <p>6.2 Hệ thống lưu câu hỏi xuống cơ sở dữ liệu</p> <p>7. Nhấn nút huỷ: quay lại trang quản lý câu hỏi</p>
Alternative flow	A1: Thông báo không lưu được câu hỏi

- Activity diagram:





- Class diagram:



Quản lý câu hỏi

Giao diện

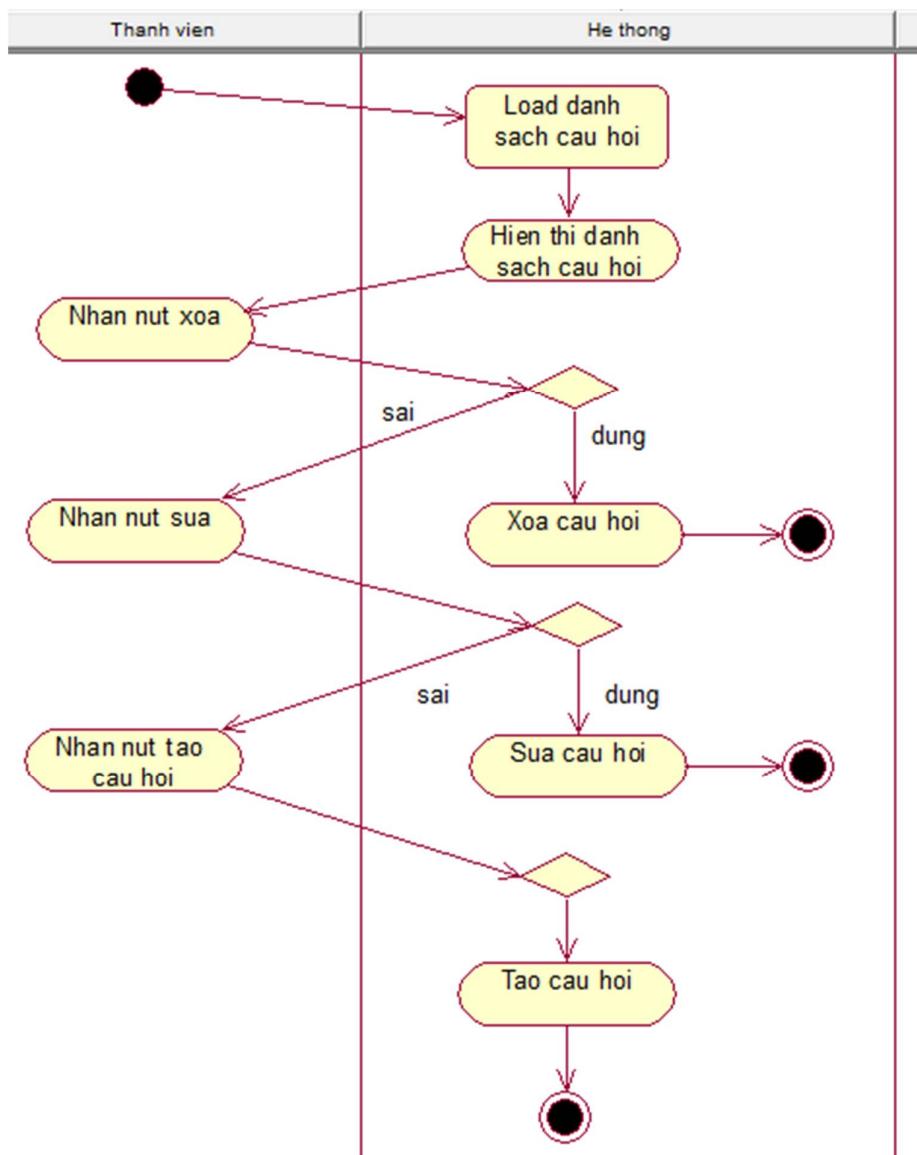
The screenshot shows the ESurvey software interface. At the top, there's a blue header bar with the ESurvey logo, the text "Tiết kiệm hơn - Hiệu quả hơn", and a navigation menu with "TRANG CHỦ", "KHẢO SÁT CỦA BẠN", and "TRỢ GIÚP". On the right side of the header, there are three rating options: "Rất tốt" (Excellent), "Tốt" (Good), and "Tạm được" (Temporary). Below the header, there's a sidebar on the left with "DANH MỤC" containing links like "Đang thiết kế", "Chờ duyệt", "Không duyệt", "Đang chạy", and "Đã kết thúc". There's also a "TÀI KHOẢN" section showing "Số ngày Vip: 3" and "Số dư tài khoản: 250000", with a "Nạp phí" button. A "TÌM KIẾM" section has a search input field and a "Tim kiem" button. The main content area is titled "DANH SÁCH CÂU HỎI" and shows a list of survey questions. The first question is "Bạn có thường xuyên uống nước ngọt có gas không" with two radio button options: "Có" and "Không". To the right are buttons for "Xóa", "Sửa", and a checked "Ánh" (Delete) button. The second question is "Bạn uống vào lúc nào" with four checkbox options: "Uống khi ăn cơm", "Uống trong khi làm việc", "Uống khi đi du lịch", and "Uống trước khi ngủ". To the right are buttons for "Xóa", "Sửa", and an unchecked "Ánh" (Delete) button. The third question is "Khi uống nước ngọt có gas bạn có cảm giác thế nào" with three checkbox options: "Sảng khoái", "Mát mẻ", and "Tinh táo". To the right are buttons for "Xóa", "Sửa", and an unchecked "Ánh" (Delete) button. At the bottom of the list are two buttons: "Tạo câu hỏi" (Create question) and "Giúp câu hỏi" (Help question).

- Đặc tả usecase:

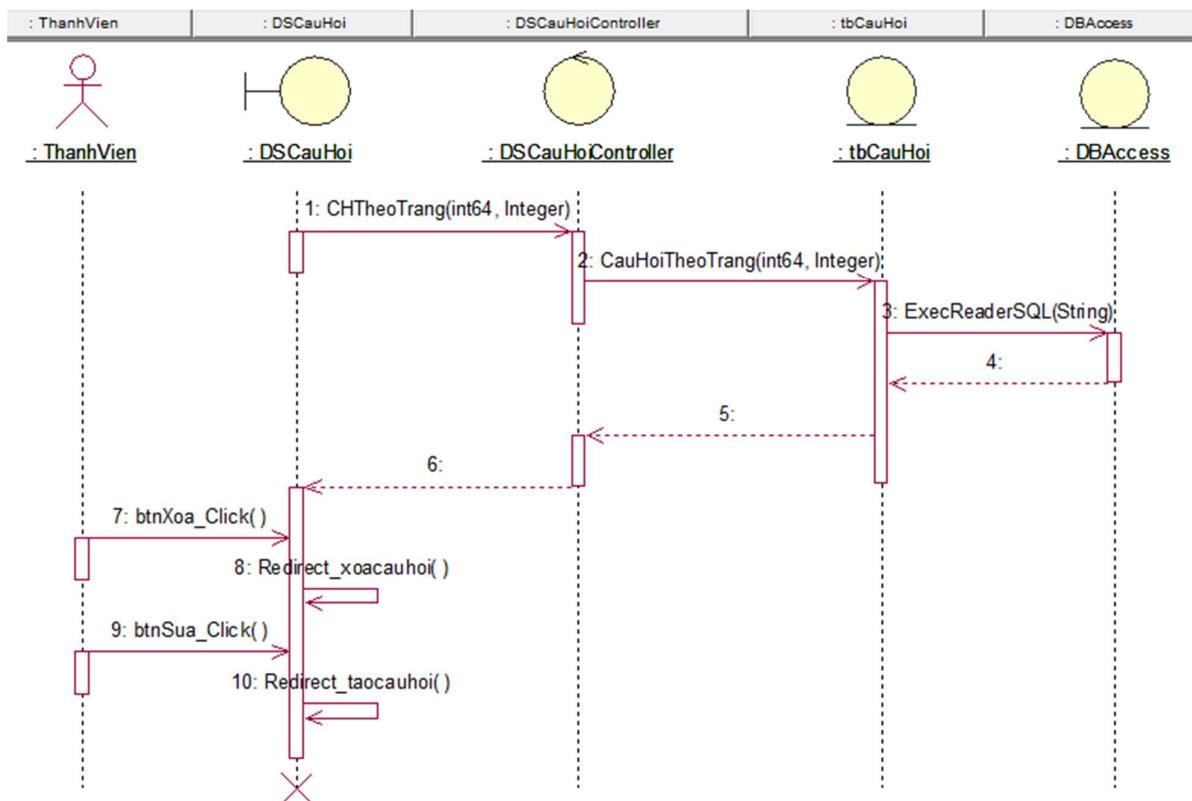
Name	Quản lý câu hỏi
Description	Thành viên quản lý tạo, cập nhật, xoá câu hỏi
Actor	Thành viên
Pre conditions	<ul style="list-style-type: none"> Đăng nhập vào tài khoản thành viên Hiển thị trang quản lý câu hỏi
Post conditions	Thông báo kết quả thao tác
Flow of events	<ol style="list-style-type: none"> Hệ thống lấy danh sách câu hỏi theo khảo sát và người tạo khảo sát. Hệ thống hiển thị danh sách câu hỏi Không hiển thị được: A1 Người dùng nhấn nút tạo câu hỏi Hệ thống chuyển sang trang tạo câu hỏi mới

	<p>7. Hệ thống hiển thị trang chỉnh sửa câu hỏi</p> <p>7.1 Không thể sửa: A2</p> <p>8. Người dùng nhấn nút xoá câu hỏi</p> <p>8.1 Không thể xóa: A3</p>
Altenative flows	<p>A1: Thông báo không hiển thị được.</p> <p>A2: Thông báo không sửa được câu hỏi</p> <p>A3: Thông báo không xóa được câu hỏi.</p>

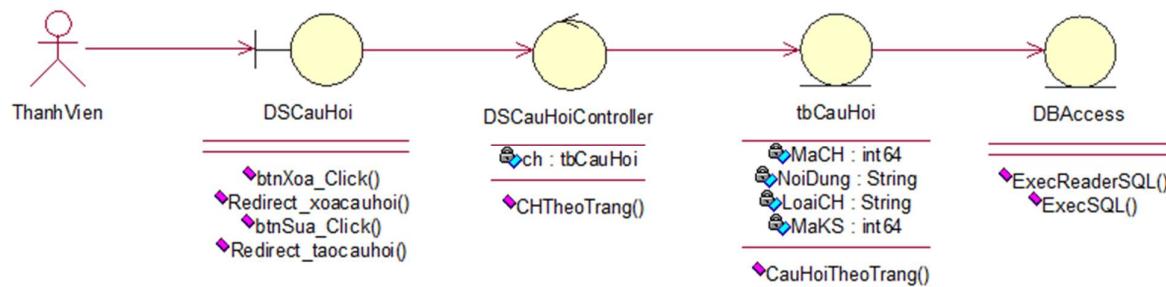
- Activity diagram:



- Sequence diagram:



- Class diagram:



Xem kết quả khảo sát

- Giao diện:

kinh Đăng xuất



Tiết kiệm hơn - Hiệu quả hơn

KHẢO SÁT TRỰC TUYẾN

TRANG CHỦ | QUẢN LÝ KHẢO SÁT | TRỢ GIÚP

Rất tốt
 Tốt
 Tạm được

QUẢN LÝ KHẢO SÁT

- [Đang thiết kế](#)
- [Chờ duyệt](#)
- [Không duyệt](#)
- [Đang chạy](#)
- [Đã kết thúc](#)

CHI TIẾT TRẢ LỜI

Nội dung trả lời	Số người chọn	
1 bạn có dự định mua thêm không	có không	3 4
2 bạn có loại hàng điện máy nào trong nhà	máy lạnh máy giặt tủ lạnh tivi	1 1 2 4
3 bạn sẽ mua j	đầu đĩa tivi lcd máy tính xách tay	1 4 2

TÀI KHOẢN
 Số ngày Vip: 3
 Số dư tài khoản: 250000
[Nạp phí](#)

TÌM KIẾM
 Tên khảo sát:
[Tim kiếm](#)

[Trang chủ](#) | [Quy định](#) | [Hướng dẫn](#) | [Sơ đồ trang](#)

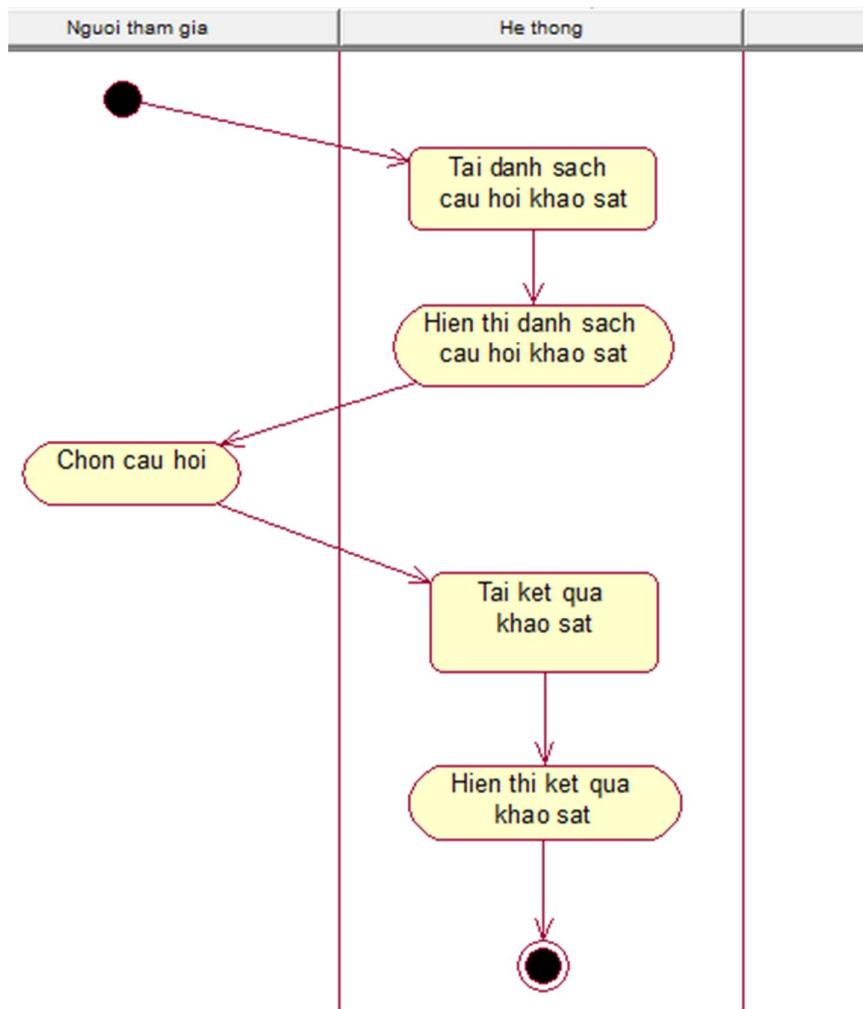
Bản quyền ESurvey 2011

- Đặc tả usecase:

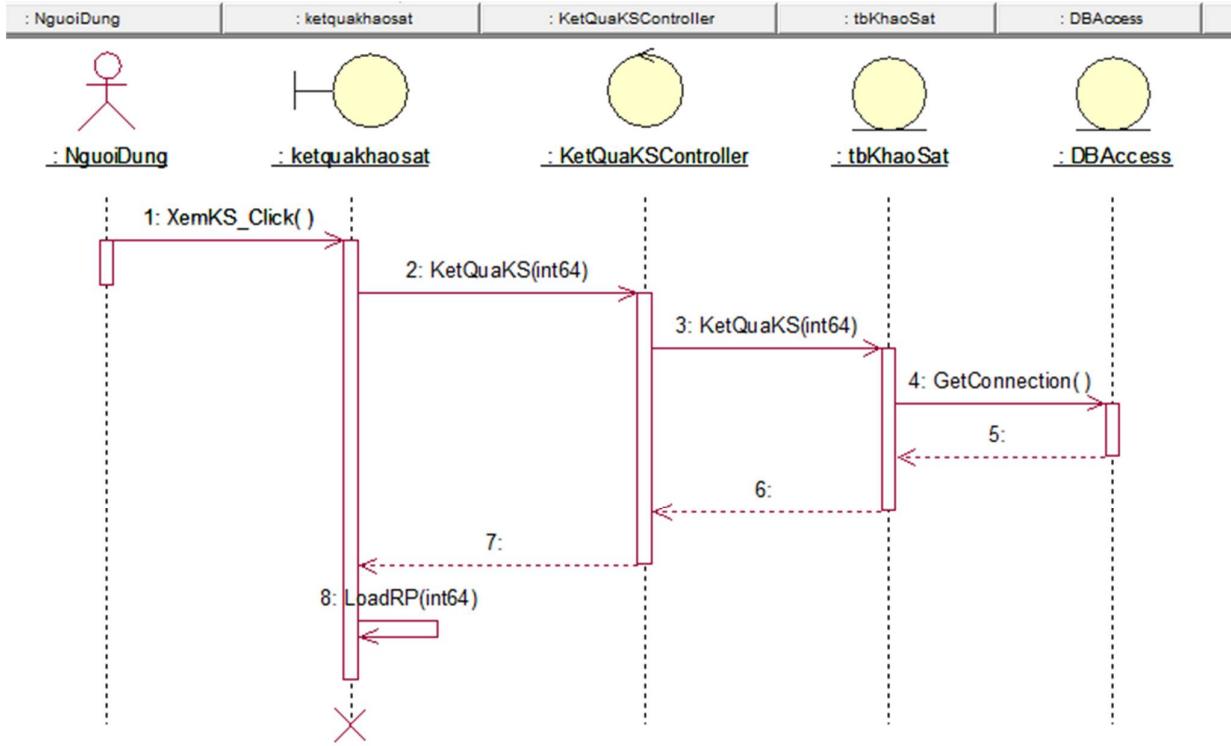
Name	Xem kết quả khảo sát
Description	Người dùng xem kết quả khảo sát đã kết thúc
Actor	Người dùng
Pre conditions	<ul style="list-style-type: none"> • Đăng nhập vào tài khoản người dùng • Hiển thị trang xem kết quả
Post conditions	Hiển thị kết quả khảo sát
Flow of events	<ol style="list-style-type: none"> 1. Hệ thống tải lên danh sách các khảo sát 2. Không tải lên được: A1 3. Hệ thống hiển thị danh sách các khảo sát 4. Người dùng chọn tên khảo sát 5. Hệ thống tải lên danh sách các câu hỏi 6. Không tải được câu hỏi: A2 7. Hệ thống hiển thị danh sách các câu hỏi 8. Hệ thống hiển thị câu trả lời và tỉ lệ

	10. Nhấn nút xem chi tiết 11. Không hiển thị được: A3 12. Hệ thống hiển thị chi tiết kết quả khảo sát của câu hỏi đã được chọn.
Alternative flows	A1: Thông báo lên trang web không tải được khảo sát A2: Thông báo không tải được câu hỏi A3: Thông báo không hiển thị được chi tiết.

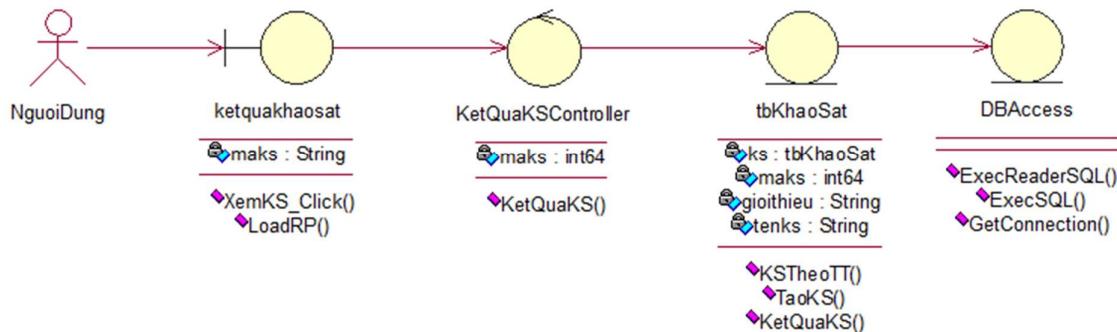
- Activity diagram:



- Sequence diagram:



- Class diagram:



Tham gia khảo sát

- Giao diện

The screenshot shows the ESurvey website interface. At the top right is a 'Đăng nhập' (Login) button. The main title 'KHẢO SÁT TRỰC TUYẾN' is displayed prominently. On the right side, there is a feedback scale with three options: 'Rất tốt' (Very good), 'Tốt' (Good), and 'Tạm được' (Temporary). The left sidebar contains a 'ĐĂNG NHẬP' (Login) form with fields for 'Tài khoản:' and 'Mật khẩu:', a 'Lưu mật khẩu.' (Save password) checkbox, and a 'Đăng nhập' (Login) button. Below it is a 'TÌM KIẾM' (Search) section with a 'Tên khảo sát:' (Survey name) input field and a 'Tim kiếm' (Search) button. The main content area is titled 'THAM GIA KHẢO SÁT' (Participate in the survey). It displays a question: 'Bạn có thường xuyên uống nước ngọt có gas không?' (Do you often drink sugary carbonated water?) with two radio button options: 'Có' (Yes) and 'Không' (No). Below this are four checkboxes for 'Bạn uống vào lúc nào?' (When do you drink?): 'Uống khi ăn cơm' (When eating), 'Uống trong khi làm việc' (When working), 'Uống khi đi du lịch' (When traveling), and 'Uống trước khi ngủ' (Before sleeping). Another question asks 'Khi uống nước ngọt có gas bạn có cảm giác thế nào?' (How do you feel when drinking sugary water?) with three checkbox options: 'Sảng khoái' (Refreshing), 'Mát mẻ' (Cooling), and 'Tinh táo' (Energetic). A blue 'Tiếp tục' (Continue) button is located at the bottom of the survey section.

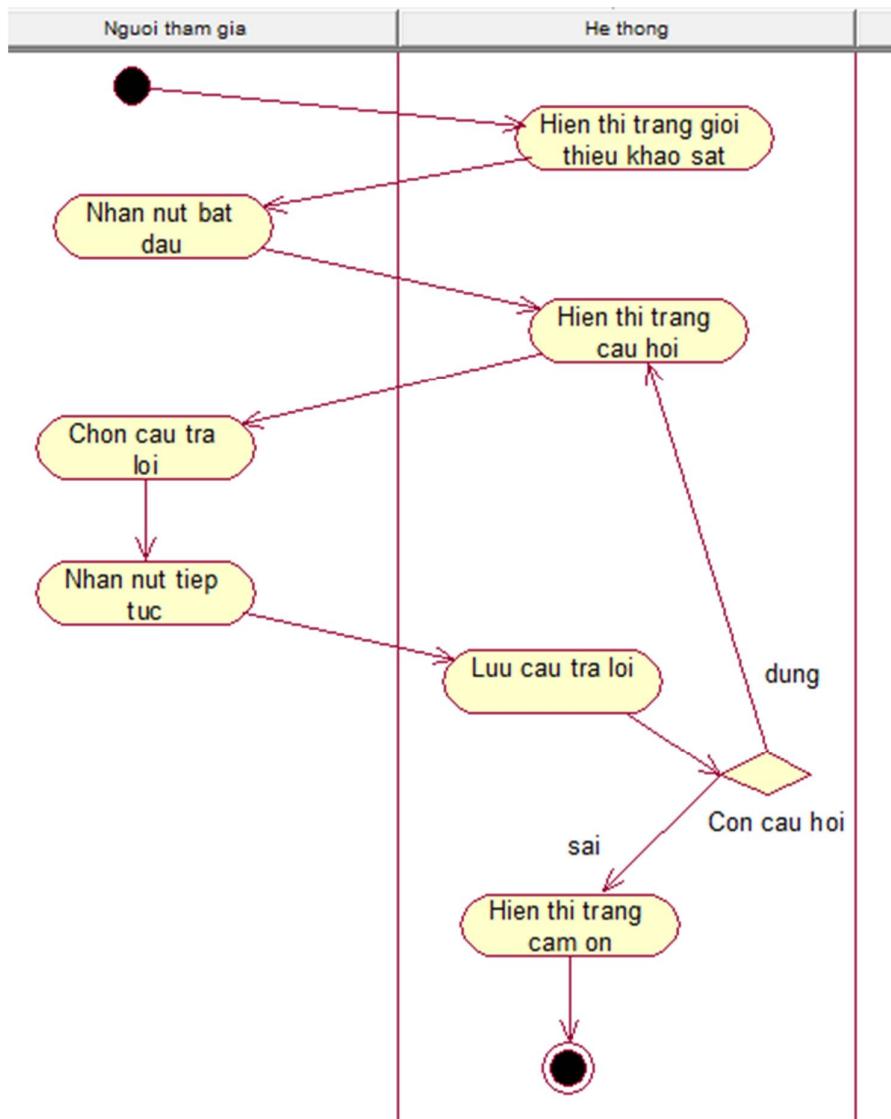
- **Đặc tả usecase:**

Name	Tham gia khảo sát
Description	Người truy cập web tham gia khảo sát
Actor	Người dùng web
Pre conditions	<ul style="list-style-type: none"> • Hiển thị trang tham gia khảo sát
Post conditions	Thông báo kết quả hoàn thành khảo sát
Flow of events	<ol style="list-style-type: none"> 1. Hệ thống hiển thị trang giới thiệu khảo sát 2. Người dùng nhấn nút bắt đầu 3. Hệ thống hiển thị trang câu hỏi đầu tiên 4. Hệ thống tải danh sách câu hỏi và trả lời 5. Người dùng chọn câu trả lời 6. Nhấn nút tiếp tục 7. Hệ thống lưu câu trả lời của người dùng xuống 8. Không lưu được: A2 9. Hiển thị trang cảm ơn người tham gia khảo sát

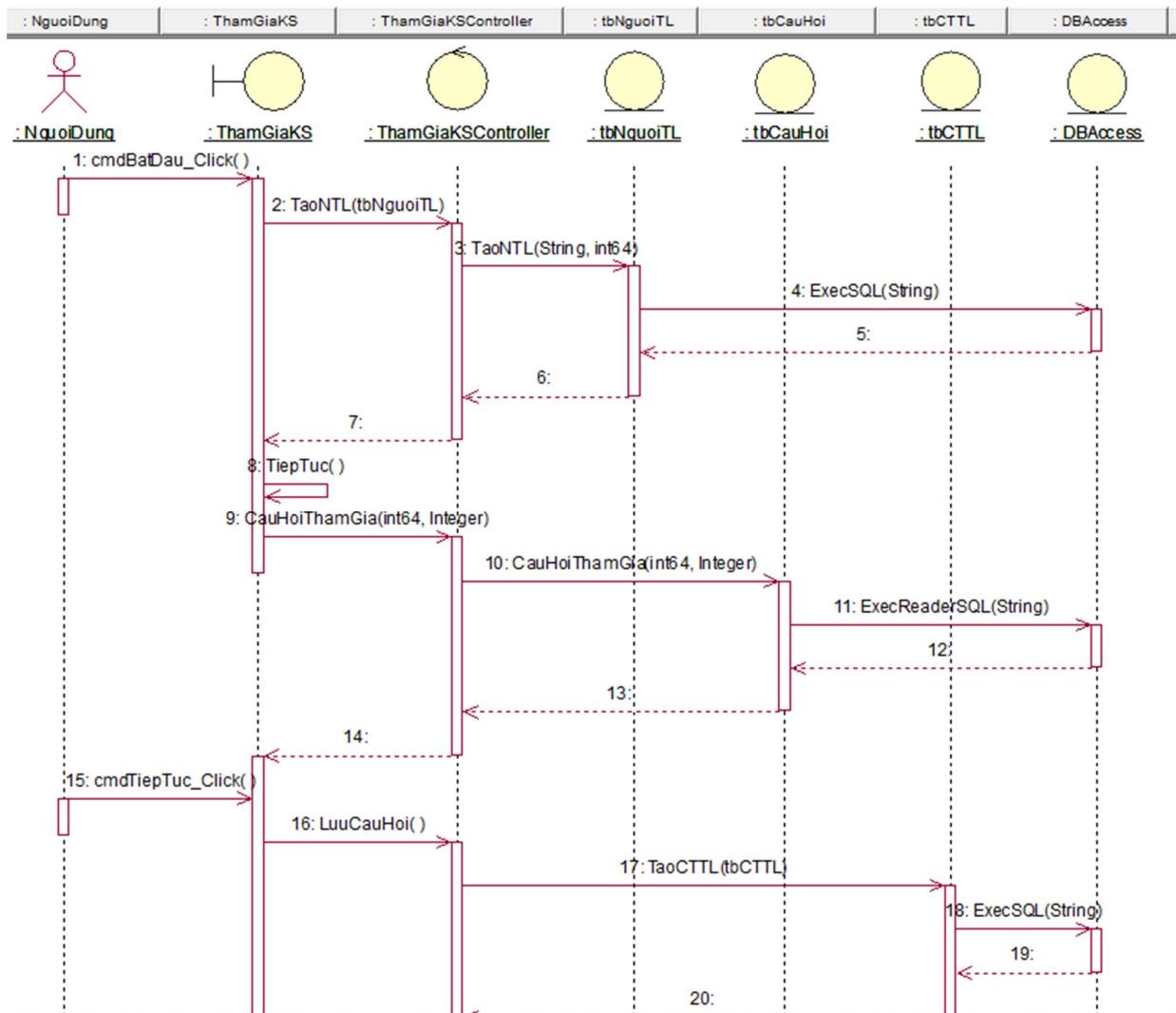
Alternative flows

A1: Thông báo không lưu được dữ liệu

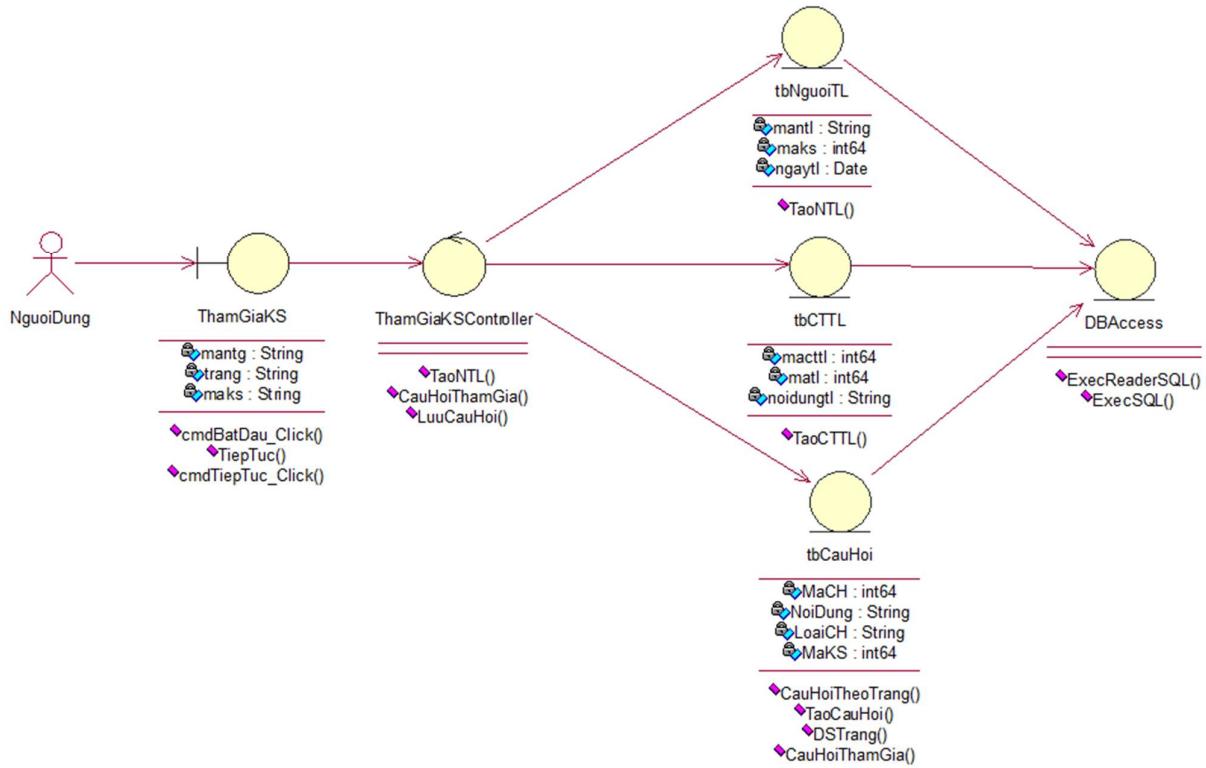
- Activity diagram:



- Sequence diagram:

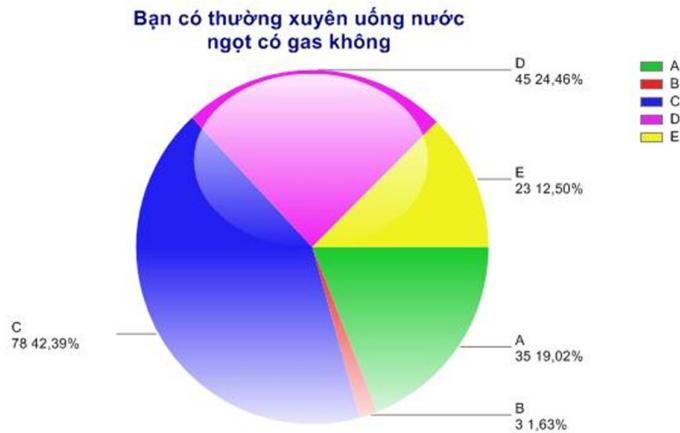


- Class diagram:



Xem biểu đồ

- Giao diện

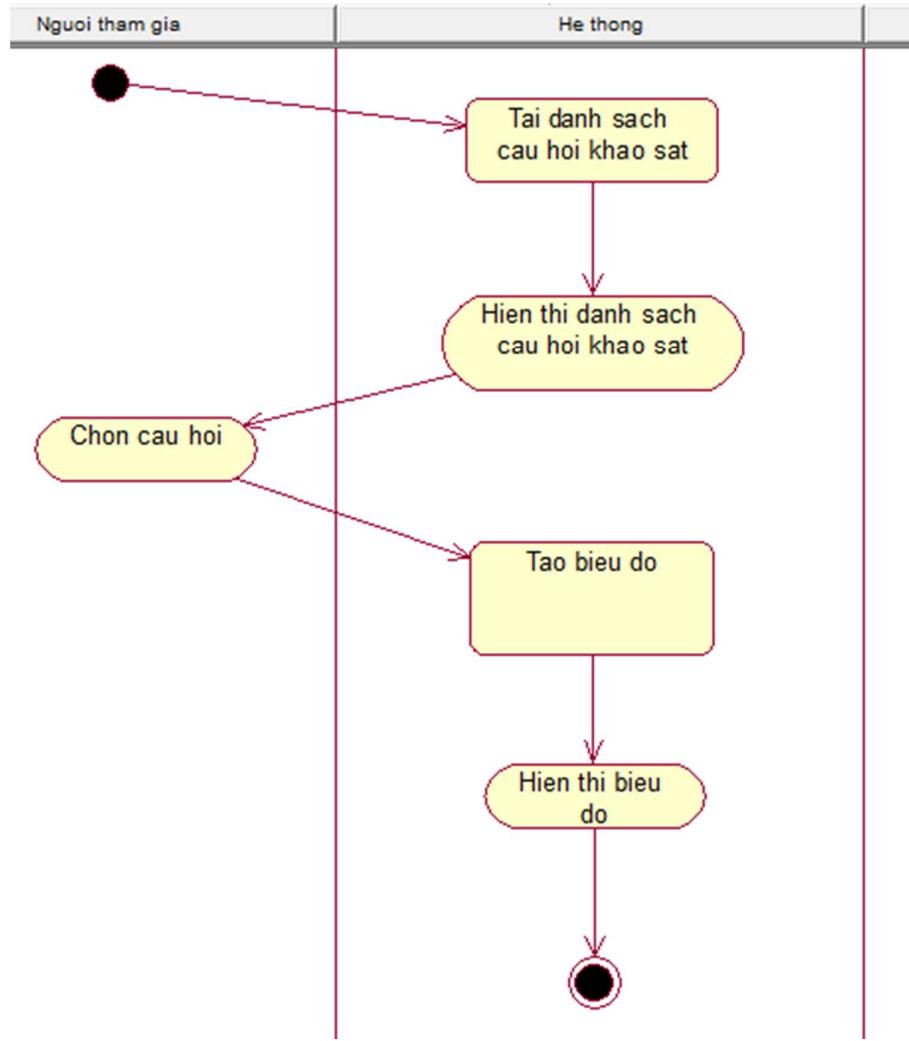


- ĐẶC TẢ USECASE:

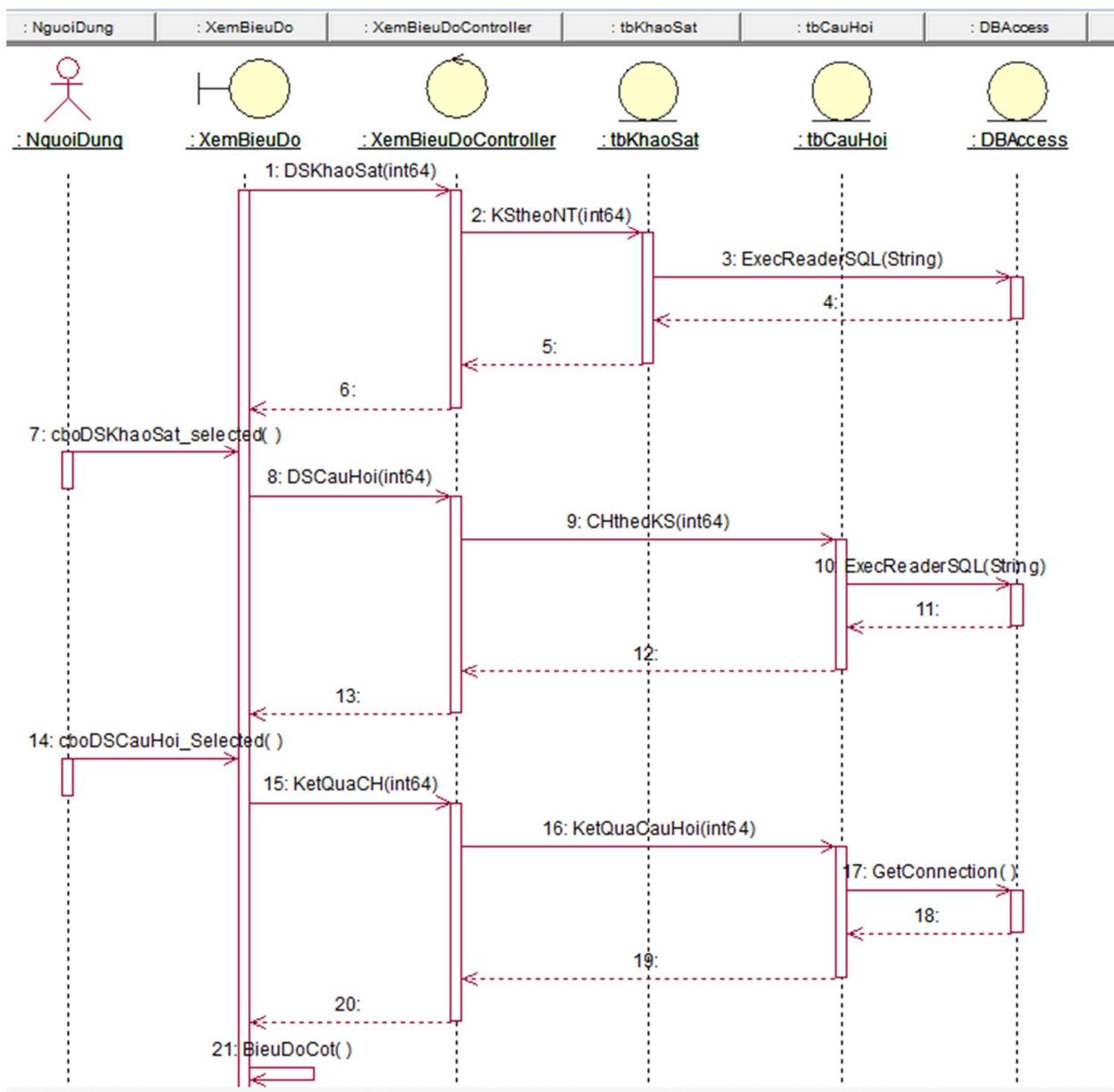
Name	Xem biểu đồ khảo sát
Description	Người dùng xem biểu đồ kết quả khảo sát đã kết thúc
Actor	Người dùng

Pre conditions	<ul style="list-style-type: none"> Đăng nhập vào tài khoản người dùng Hiển thị trang xem kết quả
Post conditions	Hiển thị biểu đồ kết quả khảo sát
Flow of events	<ol style="list-style-type: none"> Hệ thống tải lên danh sách các khảo sát Không tải lên được: A1 Hệ thống hiển thị danh sách các khảo sát Người dùng chọn tên khảo sát Nhấn nút xem biểu đồ Hệ thống tạo biểu đồ Không tạo được biểu đồ: A2 Hệ thống hiển thị biểu đồ lên trang web
Alternative flows	A1: Thông báo không tải dữ liệu lên được A2: Thông báo không tạo được biểu đồ

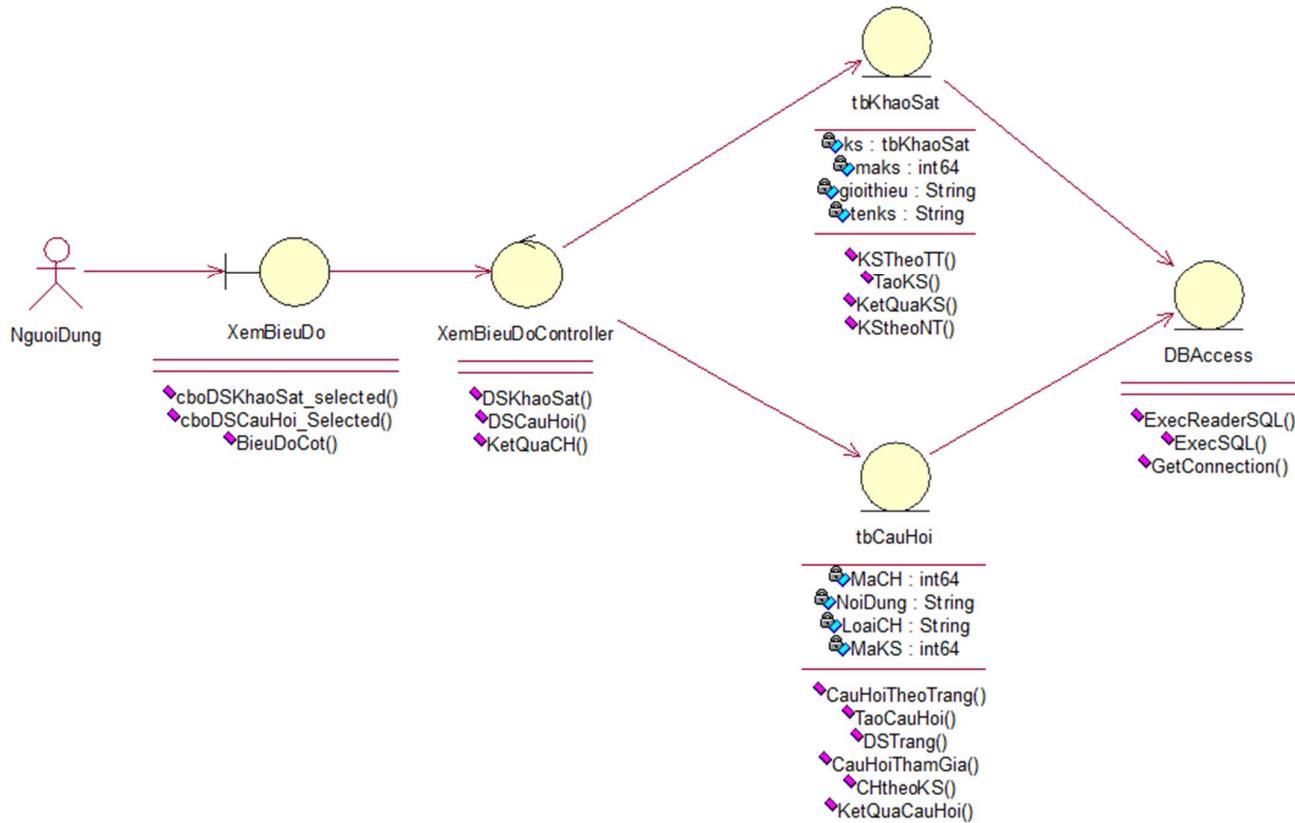
- Activity diagram:



- Sequence diagram:



- Class diagram:



Tìm kiếm khảo sát

- Giao diện

Đăng nhập

ESurvey
Tiết kiệm hơn - Hiệu quả hơn

KHẢO SÁT TRỰC TUYẾN

TRANG CHỦ | KHẢO SÁT CỦA BẠN | TRỢ GIÚP

TÌM KIẾM KHẢO SÁT

Tên khảo sát	Ngày Tạo	Loại	Tùy chọn
Khảo sát về nước giải khát THP	12/08/2011	Loại ks	Tham gia
Khảo sát về nước hoa	12/08/2011	vip	Tham gia

Tên khảo sát: nước

Tìm kiếm

Page 1 of 1 (2 items) < [1] >

ĐĂNG NHẬP

Tài khoản: _____
Mật khẩu: _____
 Lưu mật khẩu.

TÌM KIẾM

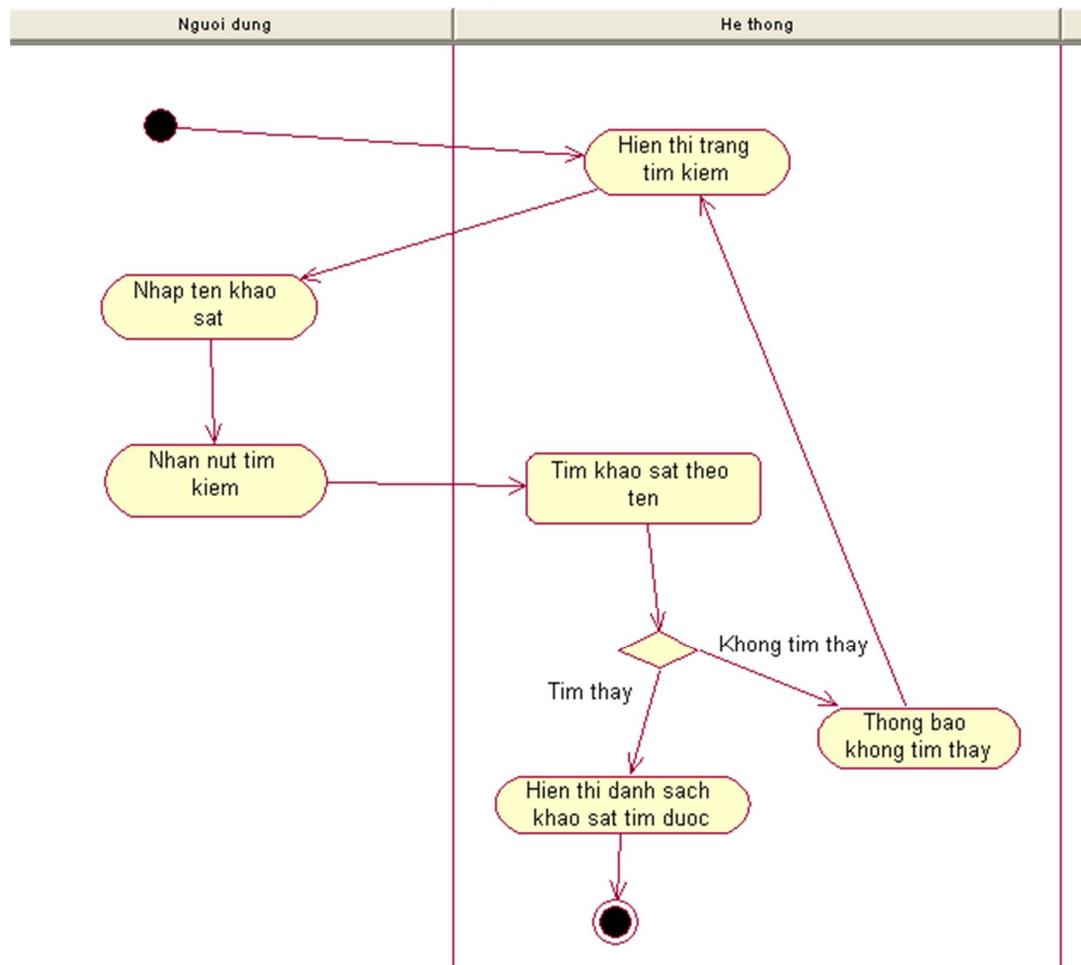
Tên khảo sát: _____

Trang chủ | Quy định | Hướng dẫn | Sơ đồ trang

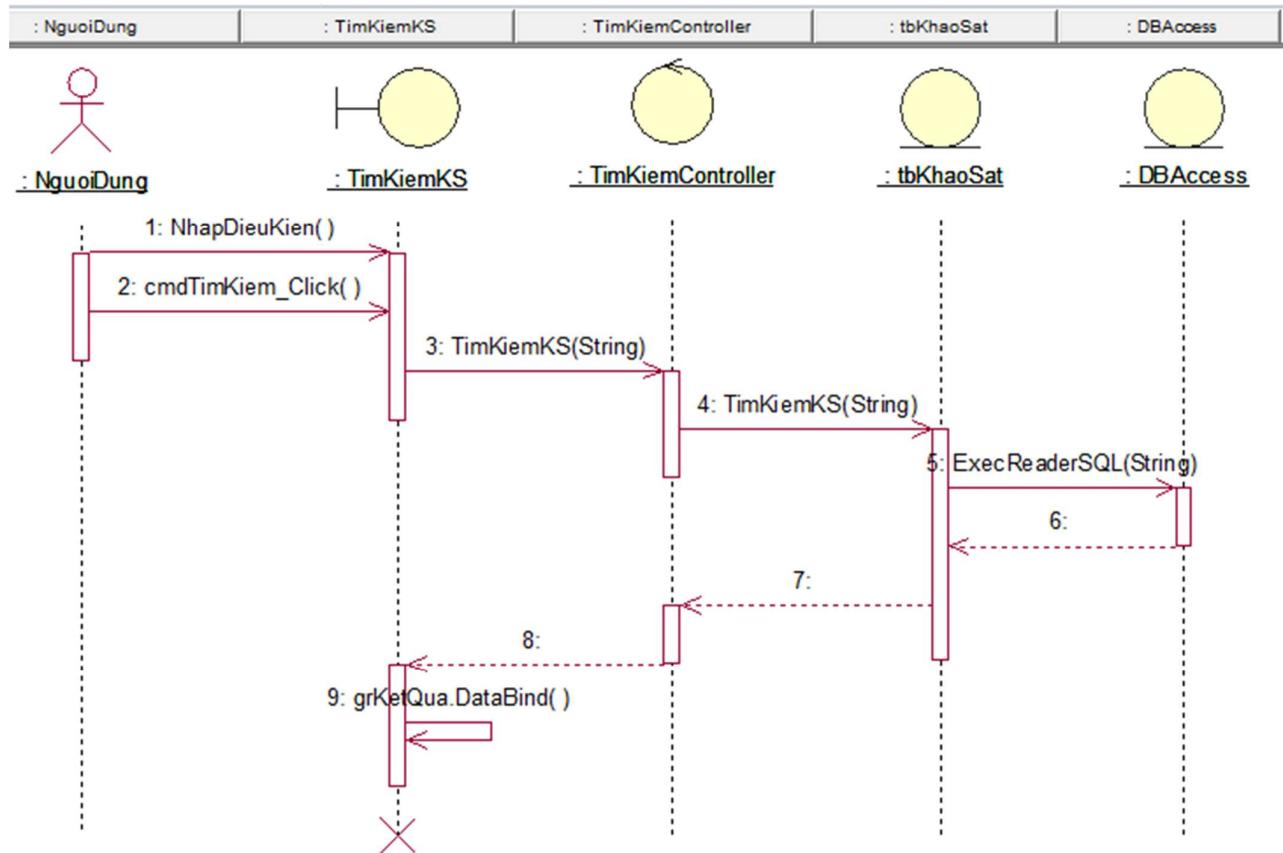
Bản quyền ESurvey 2011

Name	Tìm kiếm khảo sát
Description	Người dùng tìm kiếm khảo sát
Actor	Người dùng
Pre conditions	<ul style="list-style-type: none"> • Hiển thị trang tìm kiếm khảo sát
Post conditions	Hiển thị kết quả tìm kiếm khảo sát
Flow of events	<ol style="list-style-type: none"> 1. Hệ thống tải lên danh sách các khảo sát 2. Không tải lên được: A1 3. Hệ thống hiển thị danh sách các khảo sát 4. Người dùng chọn điều kiện tìm kiếm 5. Nhập vào thông tin tìm kiếm 6. Nhấn nút tìm kiếm 7. Hệ thống tìm kiếm khảo sát trong cơ sở dữ liệu 8. Không tìm thấy khảo sát: A2 9. Hệ thống tải lên các khảo sát tìm được 10. Không tải lên được: A3 11. Hiển thị danh sách khảo sát tìm được
Alternative flows	A1: Thông báo không tải dữ liệu lên được A2: Thông báo không tìm thấy khảo sát A3: Thông báo không tải dữ liệu lên được

- Activity diagram:



- Sequence diagram:



- Class diagram:

