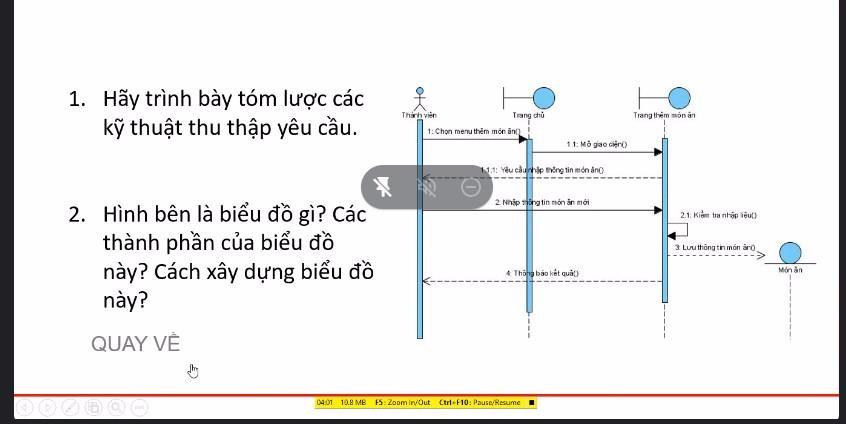
Biểu đồ trình tự:

Các kỹ thuật thu thập yêu cầu Phỏng vấn

Bảng câu hỏi

Phân tích (nghiên cứu) các tài liệu Quan sát thực tế

Phân tích thiết kế nhóm (JAD - Joint Application Design)

Boundary object : 

Entity object: 

+ Hình bên là biểu đồ (sequence diagram) biểu diễn sự tương tác giữa các đối tượng bằng việc nhấn mạnh thứ tự trao đổi thông điệp giữa các đối tượng

+ Đối tượng (Object) 

+ Đường sinh tồn (Life Line) 

+ Thông điệp (Message)

+ Vùng hoạt động (Activation) 

+ Chú thích (Notes) và ràng buộc (Constraint)

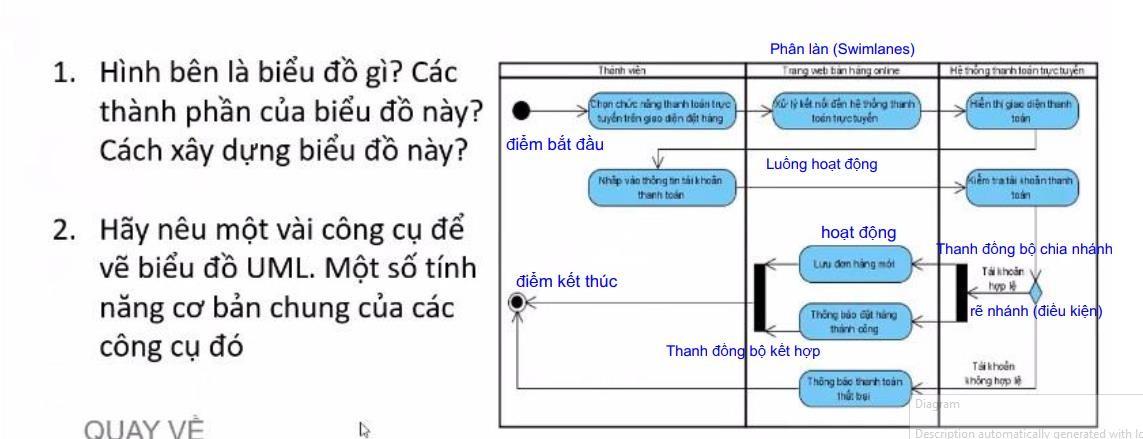
+ Tác nhân kích hoạt biểu đồ (không phải là phần tử của biểu đồ này) Để xây dựng Sequence Diagram chúng ta thực hiện các bước sau:

**Bước 1**: Xác định chức năng cần thiết kế. Bạn dựa vào Use Case Diagram để xác định xem chức

năng nào cần thiết kế.

**Bước 2**: Dựa vào Activity Diagram để xác định các bước thực hiện theo nghiệp vụ. Bước 4: Vẽ Sequence Diagarm

**Bước 5**: Cập nhật lại bản vẽ Class Diagram Biểu đồ hoạt động:



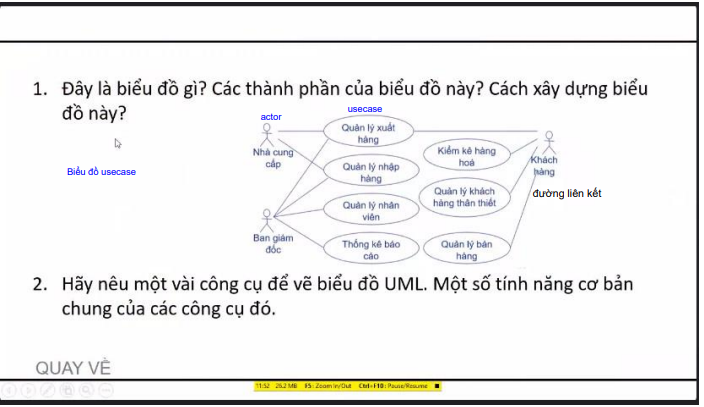
Cách xây dựng:



Các công cụ vẽ UML: draw.io, visual paradigm, start.io,…

Các tính năng cơ bản như: ngoài vẽ sơ đồ UML còn có các sơ đồ: sơ đồ mạng, sơ đồ vật lý, sơ dồ venn,…

Sơ đồ usecacse:



Cách xây dựng use case:

Giai đoạn mô hình hóa:

* Bước 1: Thực hiện thiết lập ngữ cảnh của hệ thống.
* Bước 2: Xác định các Actor.
* Bước 3: Xác định các Use Case.
* Bước 4: Định nghĩa các mối quan hệ giữa Actor và Use Case.
* Bước 5: Đánh giá các mối quan hệ đó để tìm cách chi tiết hóa.

Giai đoạn cấu trúc:

* Bước 6: Đánh giá các Use Case cho quan hệ Include.
* Bước 7: Đánh giá các Use Case cho quan hệ Extend.
* Bước 8: Đánh giá các Use Case cho quan hệ Generalization .

Giai đoạn review:

* Kiểm tra (verification): đảm bảo hệ thống đúng với tài liệu đặc tả.
* Thẩm định (validation): đảm bảo hệ thống sẽ được phát triển là thứ mà khách hàng cuối thực sự cần thiết.

Các công cụ vẽ UML: draw.io, visual paradigm, start.io,…

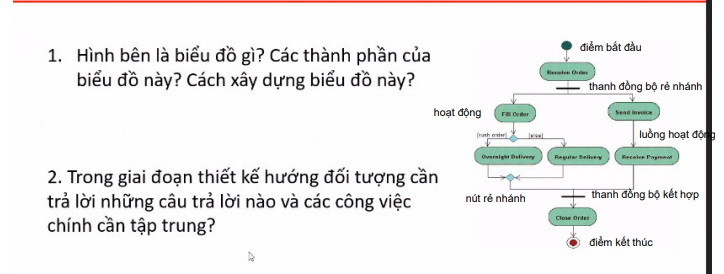
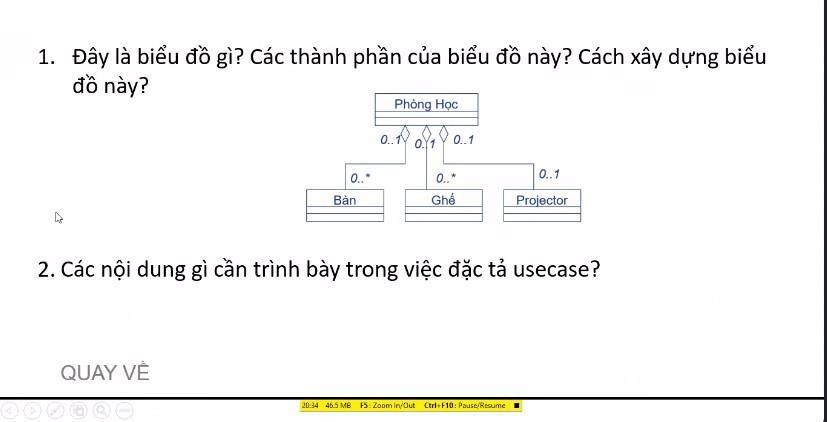
Các tính năng cơ bản như: ngoài vẽ sơ đồ UML còn có các sơ đồ: sơ đồ mạng, sơ đồ vật lý, sơ dồ venn,…

Biểu đồ hoạt động xanh lá:

* Bước 1: Xác định các nghiệp vụ cần mô tả
* Xem xét bản vẽ Use Case  để xác định nghiệp vụ nào bạn cần mô tả.
* Bước 2: Xác định trạng thái đầu tiên và trạng thái kết thúc
* Bước 3: Xác định các hoạt động tiếp theo
* Xuất phát từ điểm bắt đầu, phân tích để xác định các hoạt động tiếp theo cho đến khi gặp điểm kết thúc để hoàn tất bản vẽ này.

Trong giai đoạn thiết kế ta cần trả lời được bối cảnh của một hệ thống và kiến trúc của một hệ thống. Trong giai đoạn này,nhà phát triển hệ thống cần hiểu được tài liệu về hệ thống của họ.Sau đó sẽ khởi tạo một bản thiết kế (blueprint) của hệ thống. Bước đầu tiên trong giai đoạn này là xác định các lớp và mối liên hệ giữa chúng (inter-relationships), 6 Giai đoạn này nhà phát triển phần mềm có thể được so sánh như công việc của một kiến trúc sư. Kiến trúc sư sẽ khởi tạo 1 bản thiết kế (blueprint) hoặc mẫu (model) của tòa nhà trước khi tiến hành xây toà nhà đó

Sơ đồ lớp phòng học:



Cách xây dựng biểu đồ lớp

**Bước 1: Tìm kiếm các Classes dự kiến**

**Bước 2: Tìm các thuộc tính và phương thức cho Class**

**Bước 3: Xây dựng các quan hệ giữa các lớp và phát hiện các lớp phát sinh**

**Nội dung cần trình bày trong đặc tả use case**

## **Summary**

* **Use Case Name:** Tên Use Case
* **Use Case ID:** Mã Use Case
* **Use Case Description:** Tóm gọn nhanh sự tương tác được thể hiện trong Use Case là gì.
* **Actor:** Những đối tượng thực hiện sự tương tác trong Use Case.
* **Priority:** Mức độ ưu tiên của Use Case so với các Use Case còn lại trong dự án.
* **Trigger:** Điều kiện kích hoạt Use Case xảy ra.
* **Pre-Condition:** Điều kiện cần để Use Case thực hiện thành công.
* **Post-Condition:** Những thứ sẽ xuất hiện sau khi Use Case được thực hiện thành công.

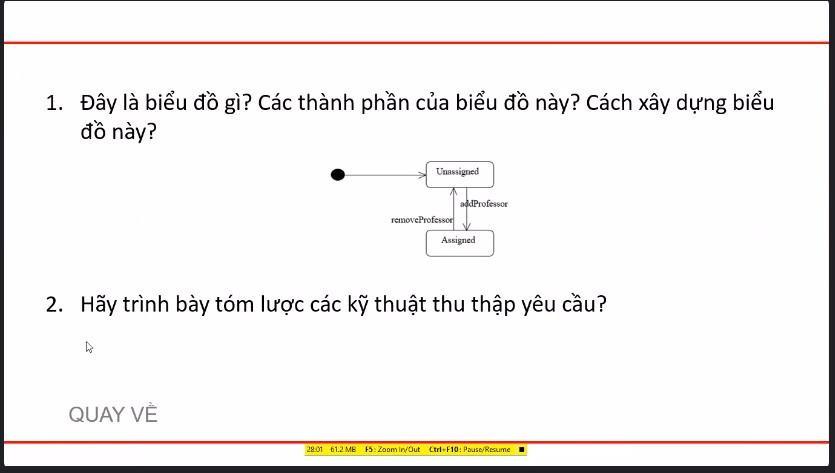
## **Flow**

* **Basic Flow:** luồng tương tác CHÍNH giữa các Actor và System để Use Case thực hiện thành công.
* **Alternative Flow:** luồng tương tác THAY THẾ giữa các Actor và System để Use Case **thực hiện thành công.**
* **Exception Flow:** luồng tương tác NGOẠI LỆ giữa các Actor và System mà Use Case **thực hiện thất bại.**

## **Additional Information**

* **Business Rule:** các quy định về mặt Business mà hệ thống bắt buộc phải nghe theo, làm theo.
* **Non-Funtional Requirement:** Vì Use Case chỉ dùng để thể hiện Functional Requirement, nên phải bổ sung các yêu cầu về Non-Functional ở đây luôn.

Biểu đồ trạng thái:



♣ B1: Đọc đặc tả phần mềm, hoặc đặc tả chi tiết dựa vào nghiệp vụ hệ thống

♣ B2: Xác định các trạng thái có thể của lớp

♣ B3: Xác định các chuyển trạng thái, và sự kiện, điều kiện chuyển, hành động để chuyển giữa các trạng thái.

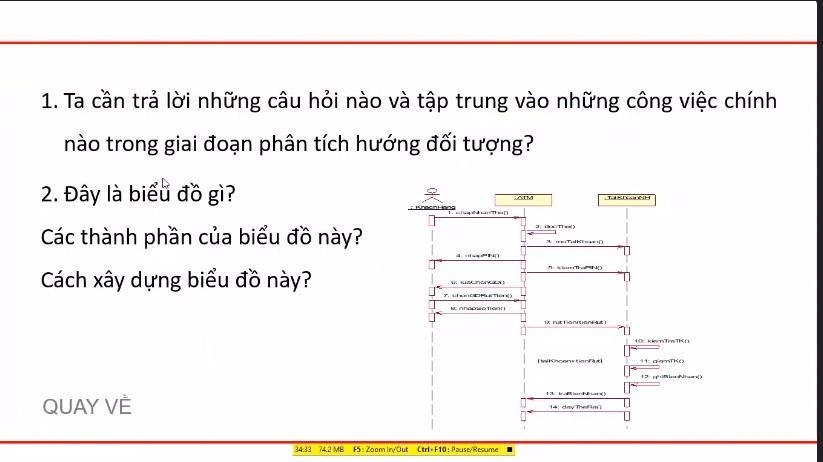
Các kỹ thuật thu thập yêu cầu Phỏng vấn

Bảng câu hỏi

Phân tích (nghiên cứu) các tài liệu Quan sát thực tế

Phân tích thiết kế nhóm (JAD - Joint Application Design)

Biểu đồ trình tự màu đỏ:



Cần trả lời những câu hỏi sau :

+ Cần sử dụng những giải pháp nào? Tính khả thi của từng giải pháp ra sao?

+ Dữ liệu hiển thị và xuất ra khác nhau ở những điểm nào?

+ Chức năng và quy trình xử lý của hệ thống phải đảm bảo những yêu cầu nào?

+ Ràng buộc giữa các đối tượng trong hệ thống cần xây được dựng ra sao?

+ Thông tin đưa vào hệ thống phải như thế nào?

Công việc chính : phân tích và làm rõ các biểu đồ người dùng (use case), biểu đồ lớp (class diagram), biểu đồ trạng thái (state diagram), biểu đồ tuần tự (Sequence diagram), biểu đồ cộng tác, biểu đồ thành phần,biểu đồ triển khai hệ thống, pha phân tích, pha thiết kế, biểu đồ hoạt động. Giai đoạn phân tích mô tả về vấn đề của dự án cần giải quyết. Các mô tả phải hoàn thiện,nhất quán…

Giai đoạn này xem xét và đưa ra giải pháp cho hệ vấn đề của dự án.Nói một cách tổng quát, phân tích là giai đoạn người sử dụng và nhà phát triển (developer) hệ thống cùng nhau đi đến sự nhất quán chung về hiểu biết của hệ thống.

Kết thúc giai đoạn thiết kế, người dùng và nhà phát triển sẽ thống nhất với nhau về các chức năng của hệ thống (system functions).

Nhà phát triển sẽ cố gắng có càng nhiều thông tin về vấn đề càng tốt. Sau đó sẽ xác định các yêu cầu quan trọng (critical requirements)

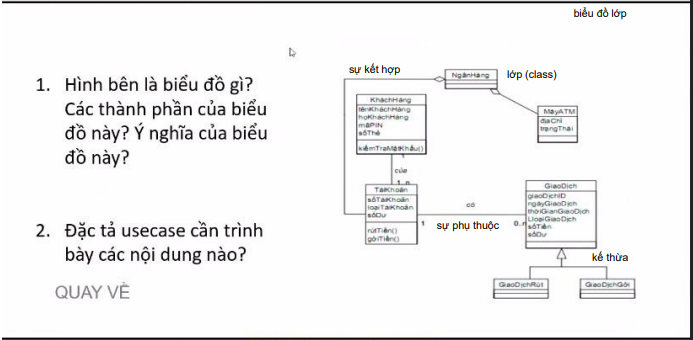
**Bước 1**: Xác định chức năng cần thiết kế. Bạn dựa vào Use Case Diagram để xác định xem chức

năng nào cần thiết kế.

**Bước 2**: Dựa vào Activity Diagram để xác định các bước thực hiện theo nghiệp vụ. Bước 4: Vẽ Sequence Diagarm

**Bước 5**: Cập nhật lại bản vẽ Class Diagram

Biểu đồ lớp màu trắng:



**Nội dung cần trình bày trong đặc tả use case**

## **Summary**

* **Use Case Name:** Tên Use Case
* **Use Case ID:** Mã Use Case
* **Use Case Description:** Tóm gọn nhanh sự tương tác được thể hiện trong Use Case là gì.
* **Actor:** Những đối tượng thực hiện sự tương tác trong Use Case.
* **Priority:** Mức độ ưu tiên của Use Case so với các Use Case còn lại trong dự án.
* **Trigger:** Điều kiện kích hoạt Use Case xảy ra.
* **Pre-Condition:** Điều kiện cần để Use Case thực hiện thành công.
* **Post-Condition:** Những thứ sẽ xuất hiện sau khi Use Case được thực hiện thành công.

## **Flow**

* **Basic Flow:** luồng tương tác CHÍNH giữa các Actor và System để Use Case thực hiện thành công.
* **Alternative Flow:** luồng tương tác THAY THẾ giữa các Actor và System để Use Case **thực hiện thành công.**
* **Exception Flow:** luồng tương tác NGOẠI LỆ giữa các Actor và System mà Use Case **thực hiện thất bại.**

## **Additional Information**

* **Business Rule:** các quy định về mặt Business mà hệ thống bắt buộc phải nghe theo, làm theo.
* **Non-Funtional Requirement:** Vì Use Case chỉ dùng để thể hiện Functional Requirement, nên phải bổ sung các yêu cầu về Non-Functional ở đây luôn.