*Câu 1: Yêu cầu phần mềm là gì? Lấy 2 ví dụ về yêu cầu phần mềm. Vai trò của yêu cầu phần mềm trong một dự án phần mềm?*

* Yêu cầu phần mềm (Software requirements) là những mô tả về những cái mà hệ thống cần phải làm:
  + Những chức năng (dịch vụ) mà hệ thống phải cung cấp.
  + Những ràng buộc đối với hệ thống.
* 2 ví dụ về yêu cầu phần mềm:
  + Hệ thống cho phép thêm một sản phẩm mới. (yêu cầu người dùng)
  + Hệ thống cho phép thêm một sản phẩm mới gồm các trường thông tin: tên sản phẩm, giá, kích thước,... Hệ thống tự động tạo mã sản phẩm theo mã danh mục sản phẩm. (yêu cầu hệ thống)
* Vai trò của yêu cầu phần mềm:

*Câu 2: Phân tích yêu cầu phần mềm là gì? Trình bày mục đích của các hoạt động yêu cầu phần mềm.*

* Phân tích yêu cầu phần mềm (Software requirements analysis) :
  + Chỉ định các đặc tính hoạt động của phần mềm
  + Cho biết giao diện của phần mềm với các phần tử hệ thống khác.
  + Thiết lập các ràng buộc mà phần mềm phải đáp ứng.
* Mục đích:
  + Xác định rõ các yêu cầu của phần mềm cần phát triển.
  + Tài liệu mô tả yêu cầu phải vừa dễ hiểu với người dùng vừa chặt chẽ để làm cơ sở cho việc lập kế hoạch.
  + Do đó yêu cầu thường được mô tả ở nhiều mức chi tiết khác nhau, nhiều giai đoạn khác nhau.

*Câu 3: Yêu cầu phi chức năng là gì? Lấy ví dụ 2 yêu cầu phi chức năng.*

* Yêu cầu phi chức năng (Non-functional requirements) là những ràng buộc (constraints) đối với các chức năng (dịch vụ) mà hệ thống cung cấp.
* Các ràng buộc như:
  + Về thời gian
  + Về hiệu năng
  + Về quy trình phát triển phần mềm.
* Yêu cầu phi chức năng thường được áp dụng toàn hệ thống hơn là từng chức năng riêng lẻ.
* Ví dụ:
  + Hệ thống phải dễ sử dụng.
  + Hệ thống phải hoạt động tốt trên Firefox x.x, Chrome x.x,…
  + Hệ thống phải phản hồi trong vòng < 1s đối với tất cả các yêu cầu tìm kiếm sách.

*Câu 4: Một số câu hỏi giúp phát hiện ra những người tham gia dự án phần mềm.*

* Xác định các bên liên quan : “Bạn nghĩ tôi nên nói chuyện với ai khác”
* Nhận ra nhiều quan điểm.
* Hướng tới sự cộng tác.
* Những câu hỏi đầu tiên:
  + Ai đứng đằng sau lời yêu cầu về công việc này?
  + Ai sẽ dùng giải pháp này?
  + Giải pháp thành công sẽ mang lại lợi ích kinh thế nào?
  + Có nguồn giải pháp nào khác mà bạn cần không?

*Câu 5: Một số điều cần lưu ý khi tiến hành thu thập yêu cầu.*

* Các cuộc họp (thực hoặc ảo) được tiến hành và tham gia bởi cả kỹ sư phần mềm và các bên liên quan khác.
* Các quy tắc chuẩn bị và tham gia được thiết lập.
* Chương trình nghị sự được đề nghị đủ chính thức để bao quát tất cả các điểm quan trọng nhưng không đủ chính thức để khuyến khích các luồng ý tưởng tự do.
* Một “người điều phối” (khách hàng, nhà phát triển hoặc người ngoài) kiểm soát cuộc họp.
* Một “cơ chế định nghĩa” (bảng tính, biểu đồ lật, dán tường hoặc diễn đàn ảo) được sử dụng.
* Mục tiêu: Xác định vấn đề, đề xuất các yếu tố giải pháp và đàm phán các phương pháp khác nhau.

*Câu 6: Danh sách các sản phẩm của hoạt động thu thập yêu cầu.*

* Tuyên bố về sự cần thiết và tính khả thi.
* Tuyên bố giới hạn về phạm vi của hệ thống hoặc sản phẩm.
* Danh sách khách hàng, người dùng và các bên liên quan khác đã tham gia vào việc thu thập các yêu cầu.
* Mô tả môi trường kỹ thuật của hệ thống.
* Danh sách các yêu cầu (tốt nhất là sắp xếp theo chức năng) và ràng buộc miền áp dụng cho từng lĩnh vực.
* Tập hợp các tình huống sử dụng (được viết bằng từ ngữ riêng của các bên liên quan) cung cấp thông tin chi tiết về việc sử dụng hệ thống hoặc sản phẩm theo điều kiện hoạt động khác nhau.

*Câu 7: Ca sử dụng là gì? Những câu hỏi cần trả lời khi xây dựng kịch bản cho ca sử dụng.*

* Use case là:
  + Tập hợp các tình huống người dùng mô tả luồng sử dụng của một hệ thống.
  + Mỗi tình huống được mô tả theo quan điểm của một “tác nhân” – một người hoặc thiết bị tương tác với phần mềm theo một cách nào đó.
* Những câu hỏi cần trả lời khi xây dựng kịch bản cho use case:
  + Ai là tác nhân chính, tác nhân phụ.
  + Mục tiêu của tác nhân là gì?
  + Điều kiện tiên quyết nào nên có trước khi câu chuyện bắt đầu?
  + Tác nhận thực hiện những nhiệm vụ hoặc chức năng chính nào?
  + Những phần mở rộng nào có thể được coi là câu chuyện được mô tả?
  + Có thể có những biến thể nào trong tương tác của tác nhân?
  + Tác nhân sẽ thu nhận, sản xuất hoặc thay đổi thông tin hệ thống nào?
  + Tác nhân sẽ phải thông báo cho hệ thống về những thay đổi của môi trường bên ngoài?
  + Tác nhân mong muốn thông tin gì từ hệ thống?
  + Tác nhân có muốn được thông báo về những thay đổi bất ngờ không?

*Câu 8: Mô hình phân tích là gì? Nêu và giải thích các loại phần tử trong mô hình phân tích.*

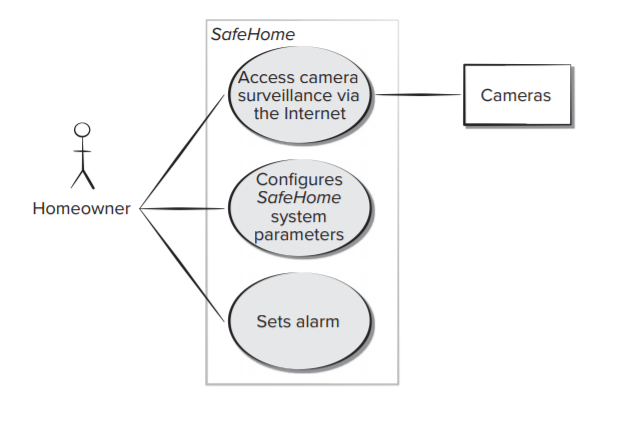
Mô hình phân tích (Analysis model) – cung cấp mô tả về các miền thông tin, chức năng và hành vi cần thiết cho một hệ thống dựa trên máy tính.

Các loại phần tử trong mô hình phân tích:

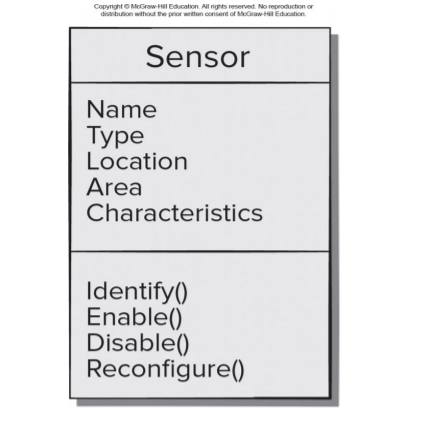
* Yếu tố dựa trên kịch bản (Scenario-based elements)
  + Loại phần tử này thể hiện quan điểm của người dùng hệ thống.
  + Các yếu tố dựa trên kịch bản là sơ đồ ca sử dụng, câu chuyện người dùng.
* Yếu tố dựa trên lớp (Class-based elements)
  + Đối tượng của loại phần tử này do hệ thống thao tác.
  + Nó xác định đối tượng, thuộc tính và mối quan hệ.
  + Sự hợp tác đang diễn ra giữa các lớp.
  + Các phần tử dựa trên lớp là sơ đồ lớp, sơ đồ cộng tác.
* Yếu tố dựa trên hành vi (Behavioral elements)
  + Thể hiện trạng thái của hệ thống và cách nó bị thay đổi bởi các sự kiện bên ngoài.
  + Các yếu tố hành vi là biểu đồ trình tự, biểu đồ trạng thái.

*Câu 9: Lấy ví dụ từng loại phần tử trong mô hình phân tích.*

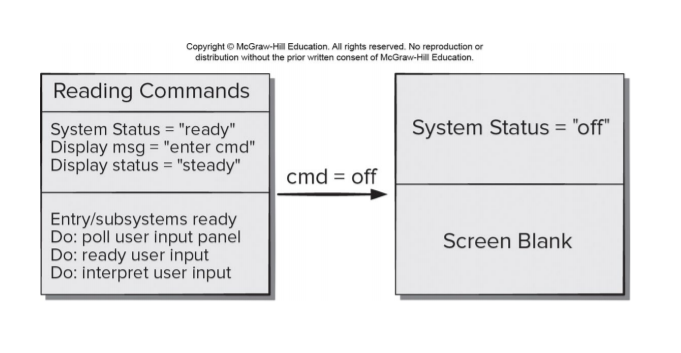
Scenario-based elements:



Class-based elements:



Behavioral elements:



*Câu 10: Những câu hỏi thường dùng trong thẩm định yêu cầu?*

* Mỗi yêu cầu có phù hợp với mục tiêu tổng thể của hệ thống / sản phẩm không?
* Tất cả các yêu cầu đã được chỉ định ở mức độ trừu tượng thích hợp chưa? Có nghĩa là, một số yêu cầu cung cấp mức độ chi tiết kỹ thuật không phù hợp ở giai đoạn này?
* Yêu cầu có thực sự cần thiết hay nó đại diện cho một tiện ích bổ sung tính năng có thể không cần thiết đối với mục tiêu của hệ thống?
* Mỗi yêu cầu có bị ràng buộc và rõ ràng không?
* Mỗi yêu cầu có ghi nhận tác giả không? Đó là một nguồn (nói chung, một các nhân cụ thể) có được ghi chú cho mỗi yêu cầu không?
* Có yêu cầu nào xung đột với các yêu cầu khác không?
* Mỗi yêu cầu có thể đạt được trong môi trường kỹ thuật sẽ chứa hệ thống hoặc sản phẩm không?
* Mỗi yêu cầu có thể kiểm tra được một khi được thực hiện không?
* Mô hình yêu cầu có phản ánh đúng thông tin, chức năng và hành vi của hệ thống được xây dựng không?
* Mô hình yêu cầu có được “phân vùng” theo cách hiển thị thông tin chi tiết hơn về hệ thống không?
* Sử dụng các mẫu yêu cầu để đơn giản hóa mô hình yêu cầu. Tất cả các mẫu đã được xác nhận đúng cách chưa?
* Tất cả các mẫu có phù hợp với yêu cầu của khách hàng không?

**Mục tiêu của việc thẩm định là xác định xem yêu cầu có thỏa mãn 4 yếu tố sau không:**

* Yêu cầu có thoả mãn nhu cầu người dùng hay không?
* Yêu cầu có mâu thuẫn với nhau hay không?
* Yêu cầu có mô tả đầy đủ tất cả các chức năng và ràng buộc hay không?
* Yêu cầu có đảm bảo các khía cạnh về kỹ thuật, kinh tế và pháp lý hay không?

*Câu 11: Các bước trong thương thuyết yêu cầu.*

* Các nhà phát triển đề xuất các giải pháp cho các yêu cầu, mô tả tác động của chúng và truyền đạt ý định của họ cho khách hàng.
* Khách hàng xem xét các giải pháp được đề xuất, tập trung vào các tính năng còn thiếu và tìm cách làm rõ các yêu cầu mới.
* Các yêu cầu được xác định là đủ tốt nếu khách hàng chấp nhận các giải pháp được đề xuất.

*Câu 12: Các hoạt động phân tích yêu cầu phần mềm.*

* Xác định nhu cầu của khách hàng.
* Đánh giá hệ thống về tính khả thi.
* Thực hiện phân tích kinh tế và kỹ thuật.
* Phân bổ chức năng cho các phần tử hệ thống.
* Thiết lập lịch trình và các ràng buộc.
* Tạo định nghĩa hệ thống.

*Câu 13: Phân tích yêu cầu có ý nghĩa gì tới kỹ sư phần mềm?*

Tầm quan trọng của phân tích yêu cầu là chúng cung cấp tầm nhìn về mục tiêu cuối cùng. Với mục tiêu trong tầm nhìn, các kỹ sư phần mềm có thể thiết lập các ưu tiên. Họ cũng có thể áp dụng kiến thức chuyên môn của mình để đưa ra các giải pháp tốt hơn nhằm đạt được những mục tiêu này.

*Câu 14: Những quy tắc trong phân tích yêu cầu phần mềm.*

* Mức độ trừu tượng phải tương đối cao – tập trung vào các yêu cầu có thể nhìn thấy trong các lĩnh vực kinh doanh hoặc vấn đề.
* Mô hình phân tích phải cung cấp cái nhìn sâu sắc về miền thông tin, chức năng và hành vi của phần mềm.
* Trì hoãn việc xem xét cơ sở hạ tầng và các mô hình phi chức năng khác cho đến sau khi thực hiện hoạt động mô hình hoá.
* Mô hình phân tích cung cấp giá trị cho tất cả các bên liên quan mô hình càng đơn giản càng tốt.

*Câu 15: Những quy tắc trong mô hình hoá yêu cầu phần mềm.*

* Nguyên tắc 1: Miền thông tin của một vấn đề phải được biểu diễn và hiểu rõ.
* Nguyên tắc 2: Các chức năng mà phần mềm này thực hiện phải được xác định.
* Nguyên tắc 3: Các hành vi của phần mềm (như là một hệ quả của sự kiện bên ngoài) phải được đại diện.
* Nguyên tắc 4: Các mô hình mô tả thông tin, chức năng và hành vi phải được phân vùng theo cách phát hiện chi tiết theo kiểu phân lớp (hoặc phân cấp).
* Nguyên tắc 5: Nhiệm vụ phân tích nên chuyển từ thông tin thiết yếu sang chi tiết thực hiện.

*Câu 16: Hoạt động phát hiện các lớp phân tích.*

* Các đối tượng mà hệ thống sẽ thao tác.
* Các hoạt động (còn được gọi là các phương thức hoặc dịch vụ) sẽ được áp dụng cho các đối tượng để thực hiện các thao tác.
* Mối quan hệ (một số thứ bậc) giữa các đối tượng.
* Sự cộng tác xảy ra giữa các lớp được xác định.

*Câu 17: Những gì trong bài toán có thể trở thành lớp phân tích?*

Kiểm tra các tình huống sử dụng được phát triển như một phần của mô hình yêu cầu và thực hiện “phân tích ngữ pháp”

* Các lớp được xác định bằng cách gạch dưới một danh từ hoặc cụm danh từ và nhập nó vào một bảng đơn giản.
* Cần lưu ý các từ đồng nghĩa.
* Nếu lớp (danh từ) được yêu cầu thực hiện một giải pháp thì nó là một phần của không gian giải pháp; ngược lại, nếu một lớp chỉ cần thiết để mô tả một giải pháp thì nó là một phần của không gian vấn đề.

Nếu các danh từ đã bị cô lập thì chúng ta tìm kiếm:

* Các thực thể bên ngoài (External entities) (ví dụ: các hệ thống, thiết bị, con người khác) sản xuất.
* Những thứ (things) (ví dụ: báo cáo, màn hình, chữ cái, tín hiệu) là một phần của miền thông tin cho vấn đề.
* Các sự kiện hoặc sự kiện xảy ra trong bối cảnh hoạt động của hệ thống.
* Vai trò (roles) của những người tương tác với hệ thống.
* Các đơn vị tổ chức có liên quan đến một ứng dụng.
* Những nơi thiết lập bối cảnh của vấn đề và chức năng tổng thể.
* Các cấu trúc (Structures) (ví dụ: cảm biến, xe bốn bánh hoặc máy tính) xác định một lớp đối tượng hoặc các lớp đối tượng liên quan.

*Câu 18: Những tiêu chí lựa chọn lớp phân tích.*

* Thông tin được lưu giữ : Lớp tiềm năng sẽ hữu ích trong quá trình phân tích chỉ khi thông tin về nó phải được ghi nhớ.
* Các dịch vụ cần thiết: Lớp tiềm năng phải có một tập hợp các hoạt động có thể xác định được, có thể thay đổi giá trị của các thuộc tính của nó theo một cách nào đó.
* Nhiều thuộc tính: Nên tập trung vào thông tin “chính”, một lớp có 1 thuộc tính duy nhất có thể được biểu diễn tốt hơn như một thuộc tính của một lớp khác.
* Các thuộc tính chung: Một tập hợp các thuộc tính có thể được xác định cho lớp tiềm năng và các thuộc tính áp dụng cho tất cả các trường hợp của lớp.
* Các thao tác thông thường: Một tập hợp các hoạt động có thể được định nghĩa cho lớp tiềm năng và các hoạt động áp dụng cho tất cả các trường hợp của lớp.
* Các yêu cầu cần thiết: Các thực thể bên ngoài xuất hiện trong không gian vấn đề và sản xuất hoặc tiêu thụ thông tin cần thiết cho giải pháp thường sẽ được định nghĩa là các lớp phân tích trong mô hình.

*Câu 19: Attribute (thuộc tính) mô tả gì? Định nghĩa thuộc tính cho lớp phân tích. Lấy ví dụ.*

Attribute (thuộc tính) – mô tả một lớp đã được chọn để đưa vào mô hình phân tích.

Định nghĩa thuộc tính cho lớp phân tích:

* Chính các thuộc tính xác định lớp – làm rõ ý nghĩa của lớp trong ngữ cảnh của không gian vấn đề.
* Để phát triển cho một tập hợp các thuộc tính có ý nghĩa cho một lớp phân tích, bạn nên nghiên cứu từng trường hợp sử dụng và chọn những “thứ” đó một cách hợp lý “thuộc” vào lớp.

Ví dụ: Class diagram for the System class – SafeHome/174

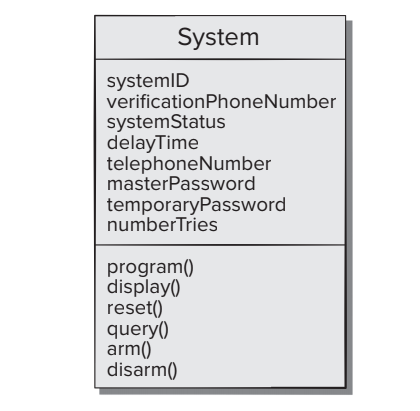
*Câu 20: Operation (thao tác) mô tả gì? Các loại thao tác khác nhau của một lớp phân tích. Lấy ví dụ.*

Các thao tác – xác định hành vi của một đối tượng.

Các loại thao tác: 4 loại lớn

* Các thao tác với dữ liệu theo một cách nào đó.
* Các thao tác thực hiện một phép tính.
* Các thao tác hỏi về trạng thái.
* Các thao tác giám sát một đối tượng để xảy ra sự kiện kiểm soát.

Ví dụ : Operation (methods)



*Câu 21: CRC là gì? CRC card là gì? Giải thích các thành phần của một CRC card.*

Class-Responsibility-Collaborator (CRC) – cung cấp một phương tiện đơn giản để xác định và tổ chức các lớp có liên quan đến các yêu cầu của hệ thống hoặc sản phẩm.

Mô hình CRC thực sự là một tập hợp các thẻ chỉ mục tiêu chuẩn đại diện cho các lớp.

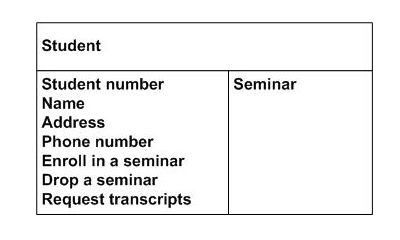
CRC card:

* Dọc trên cùng của thẻ, bạn viết tên của lớp.
* Liệt kê các trách nhiệm của lớp ở bên trái.
* Liệt kê các cộng tác viên ở bên phải.

Các thành phần của một CRC card:

* Class (lớp) – đại diện cho một tập hợp các đối tượng tương tự.
* Responsibility (trách nhiệm) – gồm có 2 phần:
  + Knowing: tương tự như thuộc tính của 1 đối tượng.
  + Doing: tương tự như hành vi của 1 đối tượng.
* Collaboration (hợp tác) – mô tả các mối quan hệ với các lớp đối tượng khác.
  + Một lớp có thể sử dụng các thao tác của chính nó để thao tác với các thuộc tính của chính nó, do đó tự hoàn thành trách nhiệm.
  + Một lớp có thể cộng tác với các lớp khác.

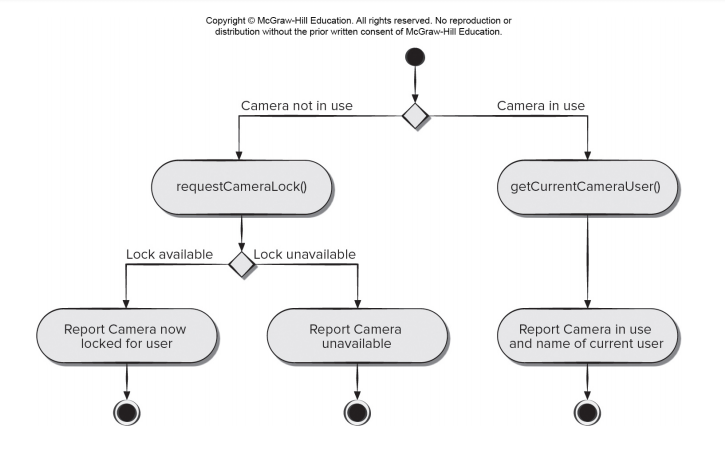
*Câu 22: Lấy ví dụ một CRC card*



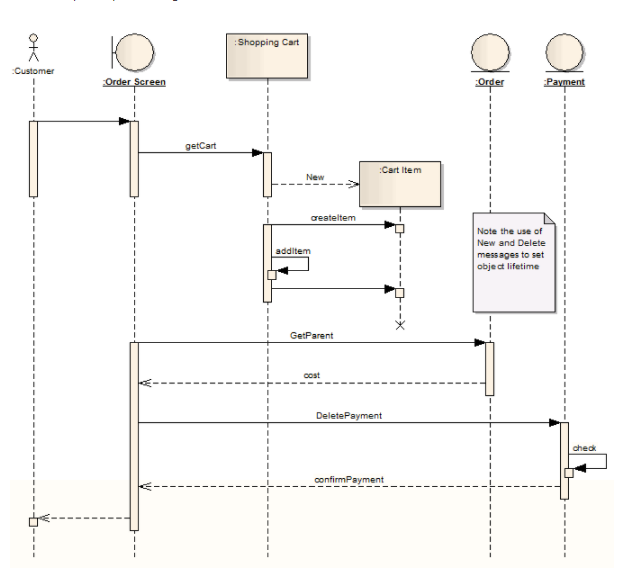
*Câu 23: Mô hình chức năng trong phân tích yêu cầu phần mềm giúp giải quyết vấn đề gì?*

* Chức năng quan sát của người dùng được ứng dụng cung cấp cho người dùng cuối.
* Các hoạt động chứa bên trong các lớp phân tích thực hiện các hành vi được liên kết với lớp.

*Câu 24:* ***Biểu đồ hoạt động*** *trong mô hình hóa chức năng.*



*Câu 25:* ***Biểu đồ tuần tự*** *trong mô hình hóa hành vi.*



*Câu 26: Các bước tạo ra mô hình hành vi.*

* Đánh giá tất cả các ca sử dụng để hiểu đầy đủ về trình tự tương tác trong hệ thống.
* Xác định các sự kiện thúc đẩy chuỗi tương tác và hiểu cách các sự kiện này liên quan đến các đối tượng cụ thể.
* Tạo một sơ đồ trình tự cho từng trường hợp sử dụng.
* Xây dựng biểu đồ trạng thái cho hệ thống.
* Xem xét mô hình hành vi về tính chính xác và nhất quán.