MỤC LỤC

[**Câu 1: Hệ thống thông tin là gì? Có những loại hệ thống thông tin nào?** 2](#_Toc99736889)

[**Câu 2: So sánh ưu và nhược điểm của hai phương pháp luận: Phát triển phần mềm hướng cấu trúc và phát triển phần mềm hướng đối tượng?** 2](#_Toc99736890)

[**Câu 3: Trình bày ngắn gọn về quy trình phát triển RUP? Nêu ưu nhược điểm của mô hình phát triển này ?** 3](#_Toc99736891)

[**Câu 4: Trình bày các mục đích sử dụng khác nhau của ngôn ngữ mô hình hoá UML? Nêu các phần tử mô hình trong UML?** 4](#_Toc99736892)

[**Câu 5: Trinh bày ý nghĩa và mục đích sử dụng của các loại biểu đồ trong ngôn ngữ mô hình hoá UML? Cơ sở nào để xác định được các loại biểu đồ đó?** 4](#_Toc99736893)

[**Câu 6 : Trình bày nội dung và ý nghĩa của các khung nhìn trong ngôn ngữ mô hình hoá UML (khung nhìn tĩnh, khung nhìn thiết kế, khung nhìn ca sử dụng, khung nhìn máy trạng thái, khung nhìn hoạt động, khung nhìn tương tác, khung nhìn triển khai, khung nhìn quản lý mô hình?** 5](#_Toc99736894)

[**Câu 7: Nêu một số vấn đề trong phát triển phần mềm hướng đối tượng? Trình bày nội dung của bộ kinh nghiệm thực tiễn để có phần mềm chất lượng tốt?** 6](#_Toc99736895)

[**Câu 8: Mục đích của hoạt động mô hình hoá nghiệp vụ là gì ? Lấy ví dụ minh hoạ?** 6](#_Toc99736896)

[**Câu 9: Mô hình miền là gì? Ý nghĩa và mục đích sử dụng của mô hình miền?** 7](#_Toc99736897)

[**Câu 10: Đặc tả bổ sung là gì? Trình bày nội dung và ý nghĩa của đặc tả bổ sung?** 7](#_Toc99736898)

[**Câu 11: Từ điển thuật ngữ là gì? Nêu nội dung và ý nghĩa của từ điển thuật ngữ?** 7](#_Toc99736899)

[**Câu 12. Mục đích của việc xác định yêu cầu là gì ? Nêu vai trò của biểu đồ ca sử dụng trong việc xác định yêu cầu khách hàng ?** 8](#_Toc99736900)

[**Câu 13. Trình bày sự khác nhau giữa một ca sử dụng và một kịch bản?** 8](#_Toc99736901)

[**Câu 14: Mục đích của hoạt động phân tích và thiết kế là gì ? Đầu vào và đầu ra cho hoạt động phân tích và thiết kế là gì ?** 8](#_Toc99736902)

[**Câu 15: Kiến trúc phần mềm là gì? Trình bày ngắn gọn về kiến trúc khung nhìn “4+1”?** 9](#_Toc99736903)

[**Câu 16: Trình bày vai trò của pha phân tích? Nêu các bước trong Phân tích hướng đối tượng?** 9](#_Toc99736904)

[**Câu 17: Mục đích của hoạt động phân tích kiến trúc là gì?** 9](#_Toc99736905)

[**Câu 18: Cơ chế phân tích là gì? Cho ví dụ?** 9](#_Toc99736906)

[**Câu 19: Mục đích của hoạt động phân tích ca sử dụng là gì?** 9](#_Toc99736907)

[**Câu 20: Lớp phân tích là gì? Nêu các kiểu mở rộng (stereotypes) của lớp phân tích? Trình bày vai trò và cách xác định các kiểu lớp: Lớp biên, lớp điều khiển, lớp thực thể?** 10](#_Toc99736908)

[**Câu 21: Các biểu đồ tương tác được tạo ra như thế nào trong hoạt động phân tích ca sử dụng? Phân biệt biểu đồ tuần tự và biểu đồ cộng tác?** 10](#_Toc99736909)

[**Câu 22: Phân biệt hai quan hệ include và extend trong biểu đồ ca sử dụng?** 10](#_Toc99736910)

[**Câu 23: Phân biệt các dạng quan hệ trong biểu đồ lớp: Association, Generalization, Aggregation, Composition, Dependency?** 10](#_Toc99736911)

[**Câu 24: Trình bày vai trò của pha thiết kế? Nêu các bước trong Thiết kế hướng đối tượng?** 11](#_Toc99736912)

[**Câu 25: Mục đích của việc xác định các phần tử thiết kế là gì?** 11](#_Toc99736913)

[**Câu 26: Hệ thống con là gì ? Phân biệt hệ thống con và gói ? Trình bày mục đích sử dụng và cách xác định các hệ thống con ?** 11](#_Toc99736914)

[**Câu 27: Cơ chế phân tích trong thiết kế hướng đối tượng dùng để làm gì ? Tại sao cần phân loại các cơ chế phân tích ?** 12](#_Toc99736915)

[**Câu 28. Mục đích thiết kế ca sử dụng là gì ?** 12](#_Toc99736916)

[**Câu 29. Mục đích của việc thiết kế lớp là gì? Các lớp được làm mịn trong thiết kế hướng đối tượng bằng cách nào ?** 12](#_Toc99736917)

[**Câu 30. Trình bày sự khác nhau giữa tiến trình (process) và luồng (thread), thao tác (operation) và phương thức (method) ?** 12](#_Toc99736918)

## **Câu 1: Hệ thống thông tin là gì? Có những loại hệ thống thông tin nào?**

- Hệ thống thông tin là một tập hợp các thành phần có liên quan với nhau, chúng thu nhập, xử lý, lưu trữ và cung cấp thông tin cần thiết để hoàn thành các tác vụ kinh doanh -> output

- Có 2 loại hệ thống thông tin:

+ Operations Support Systems (Hệ thống hỗ trợ hoạt động)

+ Management Support System (Hệ thống hỗ trợ quản lý)

## **Câu 2: So sánh ưu và nhược điểm của hai phương pháp luận: Phát triển phần mềm hướng cấu trúc và phát triển phần mềm hướng đối tượng?**

\* Phát triển phần mềm hướng cấu trúc:

- Ưu điểm:

+ đặc trưng của phương pháp hướng cấu trúc là phân chia chương trình chính thành nhiều chương trình con, mỗi chương trình con nhằm đến thực hiện một công việc xác định

+ Thiết kế dựa theo 2 hướng: hướng dữ liệu và hướng hành động.

+ tư duy phân tích thiết kế rõ ràng, chương trình sáng sủa dễ hiểu.

- Nhược điểm:

+ không hỗ trợ việc sử dụng lại

+ không phù hợp cho các phát triển các phần mềm lớn.

\* Phát triển phần mềm hướng đối tượng:

- Ưu điểm:

+ Phương pháp hướng đối tượng tập trung vào cả hai khía cạnh của hệ thống là dữ liệu và hành động

+ Hỗ trợ sử dụng lại mã nguồn

+ Phù hợp với các hệ thống lớn

- Nhược điểm:

+ tốc độ chậm

+ khó cho người mới bắt đầu

## **Câu 3: Trình bày ngắn gọn về quy trình phát triển RUP? Nêu ưu nhược điểm của mô hình phát triển này ?**

- RUP không phải là một quy trình bó hẹp cụ thể đơn nhất nhưng là một nền tảng quy trình thích ứng với sự phát triển của các tổ chức và các nhóm dự án phần mềm, tất cả sẽ chọn các yếu tố cần thiết của quy trình để phù hợp với nhu cầu, quy mô của công ty, dự án và và sản phẩm

- RUP là một liên kết các kiến trúc cơ bản với các artifact và mô tả chi tiết với các loại activity khác nhau. RUP được chứa trong sản phẩm IBM Rational Method Composer (RMC) cho phép tối ưu tiến trình

- Unified Process được thiết kế từ đặc điểm chung , quy trình phạm vi rộng lớn và RUP là một mô tả chi tiết cụ thể

- RUP hỗ trợ các hoạt động giữa các nhóm, phân chia công việc cho từng thành viên trong nhóm, trong từng giai đoạn khác nhau của quá trình phát triển phần mềm

- RUP sử dụng ký hiệu trực quan của UML và RUP được phát triển song song với UML

- RUP là kêt quả của nhiều “best pratices”, được hỗ trợ nhiều công cụ PTPM

- RUP là một sản phẩm tiến trình có thể tùy biến

\*Ưu điểm:

- Sử dụng tài nguyên một các hiệu quả

- Thường xuyên nhận được phản hồi từ các cổ đông

- Cung cấp chính xác cái mà khách hàng muốn

- Các vấn đề được phát hiện sớm trong dự án

- Hỗ trợ mô hình phát triển lặp

- Cải thiện quản lí rủ ro

\*Nhược điểm:

- Các tiến trình dự án rất phức tạp để thực hiện

- Quá trình phát triển có thể vượt quá tầm kiểm soát

- Cần các chuyên gia để có thể đáp ứng được các mục tiêu của mô hình phát triển này

- Tiến trình nặng

## **Câu 4: Trình bày các mục đích sử dụng khác nhau của ngôn ngữ mô hình hoá UML? Nêu các phần tử mô hình trong UML?**

- UML có thể được sử dụng với nhiều mục đích khác nhau:

+ UML là ngôn ngữ mô hình hoá

+ UML là ngôn ngữ trực quan hoá để hiển thị

+ UML là ngôn ngữ để đặc tả

+ UML là ngôn ngữ để xây dựng

+ UML là ngôn ngữ để làm tài liệu

- Các phần tử trong mô hình UML:

+ Phần tử cấu trúc: Phần tử cấu trúc tương ứng với các danh từ trong các mô hình UML. Chúng chủ yếu là các phần tĩnh của một mô hình, là sự tái hiện cho các thực thể khái niệm hay là vật lý. Có 7 loại phần tử cấu trúc, đó là: Lớp, Giao diện, Cộng tác, Ca sử dụng, Lớp tích cực, Thành phần và Nút

+ Phần tử hành vi: Là các phần tử mô hình UML mang khía cạnh động, nó biểu diễn các hoạt động của mô hình, các hành vi mô phỏng theo thời gian và không gian. Có 2 loại phần tử hành vi là: Tương tác và trạng thái.

+ Phần tử nhóm: là các phần tử đóng vai trò tổ chức cho các mô hình UML. Nó cung cấp cơ thể để các mô hình có thể được phân chia cũng như được nhóm gộp.

+ Phần tử chú thích: Là bộ phận chú giải của mô hình UML. Đó là các lời giải thích hay các ràng buộc thường ở dạng văn bản được áp dụng để mô tả các phần tử khác nhau trong mô hình.

## **Câu 5: Trinh bày ý nghĩa và mục đích sử dụng của các loại biểu đồ trong ngôn ngữ mô hình hoá UML? Cơ sở nào để xác định được các loại biểu đồ đó?**

* Biểu đồ lớp: bao gồm một tập các lớp, các giao diện, các collaboration, và các mối quan hệ giữa chúng. Nó thể hiện mặt tĩnh của hệ thống
* Biểu đồ đối tượng: bao gồm một tập hợp các đối tượng và mối quan hệ giữa chúng. Biểu đồ đối tượng là một thể hiện của biểu đồ lớp
* Biểu đồ use case: bao gồm một tập hợp các use case, các action và thể hiện mối quan hệ tương tác giữa các actor và use case. Nó rất quan trọng trong việc tổ chức và mô hình hóa hành vi của hệ thống.
* Biểu đồ trình tự: là 1 dạng biểu đồ tương tác, biểu diễn sự tương tác giữa các đối tượng theo thứ tự thời gian. Nó mô tả các đối tượng liên quan trong 1 tình huống cụ thể vào các bước tuần tự trong việc trao đổi các thông báo giữa các đối tượng đó để thực hiện một chức năng nào đó của hệ thống
* Biểu đồ hợp tác: gần giống như trình tự nhưng nó tập trung vào việc trao đổi qua lại thông báo giữa các đối tượng chứ không quan tâm đến thứ tự
* Biểu đồ chuyển trạng thái: nó chỉ ra một máy chuyển trạng, bao gồm các trạng thái, các bước chuyển trạng và các hoạt động của nó. Nó đặc biệt quan trọng trong việc mô hình hóa hành vi của một lớp giao diện và nó nhấn mạnh vào các đáp ứng theo sự kiện của 1 đối tượng
* Biểu đồ hạot động: là một dạng đặc biệt của biểu đồ chuyển trạng, nó chỉ ra luồng đi từ hoạt động này sang hoạt động khác trong hệ thống, nó đặc biệt quan trọng trong việc xây dựng mô hình chức năng của hệ thống
* Biểu đồ thành phần: chỉ ra cách tổ chức và sự phụ thuộc các thành phần, nó liên quan đến biểu đồ lớp, trong đó 1 thành phần thường ánh xạ tới 1 hay nhiều lớp giao diện

## **Câu 6 : Trình bày nội dung và ý nghĩa của các khung nhìn trong ngôn ngữ mô hình hoá UML (khung nhìn tĩnh, khung nhìn thiết kế, khung nhìn ca sử dụng, khung nhìn máy trạng thái, khung nhìn hoạt động, khung nhìn tương tác, khung nhìn triển khai, khung nhìn quản lý mô hình?**

* Use case view: không chỉ ra cách cấu trúc của hệ thống phần mềm mà chỉ dùng để nhìn nhận một cách tổng quát những gì mà hệ thống sẽ chung cấp, thông qua đó người dùng có thể kiểm tra xem các yêu cầu của mình đã được đáp ứng đầy đủ hay chưa hoặc có chức năng nào của hệ thống là không cần thiết
* Logic view: được dùng để xem xét các phần tử bên trong hệ thống và mối quan hệ sự tương tác giữa chúng để thực hiện các chức năng mong đợi của hệ thống
* Process view: chia hệ thống thành các tiến trình và luồng, mô tả việc đồng bộ hóa và xử lí đồng thời. Dùng cho người phát triển và tích hợp hệ thống bao gồm các biểu đồ sequence, collaboration, activity, state
* Implementation view: bao gồm các component và các file tạo nên hệ thống vật lý. Nó chỉ ra sự phụ thuộc giữa các thành phần này, cách kết hợp chúng với nhau để tạo ra một hệ thống thực thi
* Deployment view: chỉ ra cấu hình phần cứng mà hệ thống sẽ chạy trên đó. Nó thể hiện sự phụ thuộc giữa các thành phần này, cách kết hợp chúng lại với nhau để tạo ra 1 hệ thống thực thi
* Khung nhìn tĩnh: Khung nhìn tĩnh là nền tảng cơ bản của UML. Các phần tử của khung nhìn tĩnh của mô hình là các khái niệm trong miền ứng dụng, bao gồm các khái niệm thế giới thực, các khái niệm trừu tượng, các khái niệm cài đặt, các khái niệm máy tính, và nhìn chung, tất cả các khái niệm có trong hệ thống. Chẳng hạn, hệ thống quản lý đăng ký khóa học có các khái niệm như sinh viên, giảng viên, môn học, khóa học, các thuật toán về thời kháo biểu, và các trang web để đăng ký.
* Khung nhìn triển khai: Khung nhìn triển khai phản ánh cách bài trí vật lý của các nút. Nút là một nguồn lực tài nguyên tính toán như máy tính hay các thiết bị khác. Lúc thực thi, nút có thể chứa các chế tác, ác thực thể vật lý như các tệp. Quan hệ hiện thân chỉ mối liên hệ giữa các phần tử thiết kế ( các thành phần và các chế tác hiện hữu trong hệ thoogns phần mềm ). Khung nhìn triển khai giúp đánh giá hiệu năng thực thi của hệ thống, làm rõ các trình diễn dẫn đến tắc nghẽn do cách phân bổ các chế tác ( sự hiện thân của các thành phần ) trên các nút xử lý.

## **Câu 7: Nêu một số vấn đề trong phát triển phần mềm hướng đối tượng? Trình bày nội dung của bộ kinh nghiệm thực tiễn để có phần mềm chất lượng tốt?**

* Một số vấn đề trong phát triển phần mềm hướng đối tượng:

+) Phần mềm phát triển theo hướng đối tượng phải trải qua đầy đủ các pha như xác định yêu cầu, phân tích yêu cầu hướng đối tượng, thiết kế hướng đối tượng, cài đặt, kiểm thử, phát hành. Nếu thiết một trong các pha này thì phần mềm sẽ không tốt => phức tạp hơn

+) Phần mềm phát triển theo hướng đối tượng sẽ có tính modun hóa rất cao => phần mềm sẽ chậm và kích thước sẽ lớn, nếu lập trình viên không có kinh nghiệm sẽ rất khó để bảo trì, đồng thời sẽ rất khó cho những lập trình viên mới bắt đầu.

+) Thực tế, có những vấn đề mặc định sẽ được giải quyết tốt hơn nếu lập trình viên sử dụng phương pháp lập trình chức năng (Functional Programming), lập trình logic, hoặc lập trình thủ tục. Nếu ta áp dụng lập trình hướng đối tượng, có thể sẽ không đem lại hiệu quả tốt.

* Nội dung của bộ kinh nghiệm thực tiễn để có phần mềm chất lượng tốt

+) Tính đúng: Một sản phẩm thực hiện được gọi là đúng nếu nó thực hiện chính xác những chức năng đã đặc tả và thỏa mãn các mục đích công việc của khách hàng.

+) Tính kiểm thử được: Phần mềm có thể kiểm thử được là phần mềm mà nó có cách dễ dàng để có thể kiểm tra được. Đảm bảo rằng nó thực hiện đúng các chức năng dự định.

+) Tính an toàn: Tính an toàn của sản phẩm phần mềm được đánh giá thông qua:

* Có cơ chế bảo mật và bảo vệ các đối tượng do hệ thống phát sinh hoặc quản lý.
* Bản thân sản phẩm được đặt trong một cơ chế bảo mật nhằm chống sao chép trộm hoặc làm biến dạng sản phẩm đó.

+) Tính toàn vẹn: Có cơ chế ngăn ngừa việc thâm nhập bất hợp pháp vào phần mềm hay dữ liệu và ngăn ngừa việc phát sinh ra những đối tượng (dữ liệu, đơn thể...) sai quy cách hoặc mâu thuẩn với các đối tượng sẳn có. Không gây ra nhập nhằng trong thao tác. Đảm bảo nhất quán về cú pháp. Có cơ chế phục hồi lại toàn bộ hoặc một phần những đối tượng thuộc toàn bộ hoặc một phần những đối tượng thuộc diện quản lý của sản phẩm trong trường hợp có sự cố như hỏng máy, mất điện đột ngột.

+) Tính tiêu chuẩn và tính chuẩn: Sản phẩm phần mềm cần đạt được một số tiêu chuẩn tối thiểu được thừa nhận trong thị trường hoặc trong khoa học, và có thể chuyển đổi dạng cấu trúc dữ liệu riêng của hệ thống sang chuẩn và ngược lại.

+) Tính độc lập: Phần mềm cần và nên đảm bảo được tính độc lập với các đối tượng sau: Độc lập với thiết bị, Độc lập với cấu trúc của đối tượng mà sản phẩm đó quản lý, Độc lập với nội dung của đối tượng mà sản phẩm đó quản lý.

+) Khả năng phát triển mở rộng: Một phần mềm tốt luôn phải có kế hoạch phát triển và nâng cấp tính năng mới liên tục. Hãy yêu cầu nhà cung cấp nêu rõ kế hoạch phát triển những tính năng mới trong tương lai gần.

## **Câu 8: Mục đích của hoạt động mô hình hoá nghiệp vụ là gì ? Lấy ví dụ minh hoạ?**

Mô hình hóa nghiệp vụ hướng đến các mục đích sau:

* Để hiểu được cấu trúc và khía cạnh động của tổ chức trong đó hệ thống được triển khai.
* Để hiểu được vấn đề thực tại của tổ chức, xác định các cải tiến nhằm nâng cao hiệu quả của tổ chức.
* Để đảm bảo cái hiểu thống nhất về tổ chức giữa khách hàng, người dùng cuối và người phát triển.
* Để nắm bắt các yêu cầu hệ thống cần hỗ trợ cho tổ chức

Với các mục đich này, RUP đề ra các nguyên lý mô hình hóa nghiệp vụ, môt tả làm thế nào để xay dựng cái nhìn tổng quan về tổ chức mới. Để từ đó xác định được các tiến trình, vai trò và trách nhiệm của tổ chức và biểu diễn chúng bằng mô hình ca sử dụng nghiệp vụ và mô hình đối tượng nghiệp vụ.

## **Câu 9: Mô hình miền là gì? Ý nghĩa và mục đích sử dụng của mô hình miền?**

- Mô hình miền: Mô hình miền (Domain models), thiết lập các khái niệm và ngữ nghĩa cơ bản của một miền nhất định đối với nhiều đối tượng của các bên liên quan. Các mô hình miền được xây dựng dựa trên Kiến trúc tổng thể bằng cách sử dụng các mô hình Mối quan hệ thực thể hoặc Lớp UML.

- Ý nghĩa và mục đích sử dụng của mô hình miền:Mô tả các phần tử cấu trúc và các ràng buộc khái niệm của chúng trong một miền quan tâm. Mô hình miền bao gồm các thực thể khác nhau, các thuộc tính và mối quan hệ của chúng, cộng với các ràng buộc chi phối tính toàn vẹn khái niệm của các phần tử mô hình cấu trúc bao gồm miền vấn đề đó để diễn tả chính xác những điều cần có và cần phải làm thông qua mô hình.

## **Câu 10: Đặc tả bổ sung là gì? Trình bày nội dung và ý nghĩa của đặc tả bổ sung?**

* Đặc tả bổ sung: Đặc tả bổ sung giống như một bảng ghi chú, mô tả tất cả các thông tin, hành vi, dữ liệu, đối tượng, một cách chi tiết khi chúng ta đã có một cái nhìn tổng quan thông qua Usecase-Diagram.
* Nội dung và ý nghĩa: Đặc tả bổ sung gồm:
  + **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.
* **Diễn đạt chi tiết các bước, hành vi của 1 usecase.**

## **Câu 11: Từ điển thuật ngữ là gì? Nêu nội dung và ý nghĩa của từ điển thuật ngữ?**

Từ điển thuật ngữ là bộ phận của tư liệu trong phân tích thiết kế, nó là văn phạm giả hình thức mô tả nội dung của các đối tượng sự vật theo định nghĩa có cấu trúc trong luồng dữ liệu các chức năng vật lý, kho dữ liệu, luồng dữ liệu mô tả ở mức khái quát thường là tập hợp các khoản mục riêng lẻ. Các khái quát này cần được mô tả chi tiết hơn hóa đơn thông qua thông qua từ điển dữ liệu

- Mục đích là cung cấp cho người phát triển một cách hiểu thống nhất để sử dụng chúng và tạo được sự thuận tiện trong giao tiếp.

- Từ điển thuật ngữ chủ yếu được tạo ra trong giai đoạn đầu của dự án, giai đoạn đòi hỏi sự thống nhất về các thuật ngữ được dùng.

- Từ điển thuật ngữ thường được trình bày theo một định dạng riêng, tùy thuộc vào từng dự án

## **Câu 12. Mục đích của việc xác định yêu cầu là gì ? Nêu vai trò của biểu đồ ca sử dụng trong việc xác định yêu cầu khách hàng ?**

- Mục đích :

+ Để thiết lập và duy trì những thỏa thuận giữa khách hang và các bên liên quan về những gì mà hệ thống nên làm

+ Để cung cấp cho người phát triển một cái hiểu tốt hơn về các yêu cầu phần mềm

+ Để xác định biên của hệ thống

+ Để cung cấp một cơ sở cho việc lập kế hoạch những nội dung kỹ thuật trong các vòng lặp của quá trình phát triển

+ Là cơ sở cho việc đánh giá và ướng lượng chi phí và thời gian phát triển hệ thống

+ Định hình giao tiếp tương tác giữa người dung và hệ thống, qua đó làm rõ được nhu cầu và mục tiêu của người dung đối với hệ thống

- Mô hình ca sử dụng mô tả những gì mà hệ thống sẽ làm.

+ Mô hình ca sử dụng chính là một sự cam kết giữa khách hàng, người dung và người phát triển.

+ Một mặt nó cho phép khách hàng và người dung kiểm tra xem hệ thống được xây dựng có đúng như mong đợi không. Mặt khác nó cho phép người phát triển đảm bảo rằng hệ thống xây dựng là đúng với yêu caayfu của khách hàng và người dùng.

+ Mô hình ca sử dụng bao gồm các ca sử dụng và các tác nhân. Mỗi ca sử dụng trong mô hình mô tả chi tiết về những gì hệ thống làm trong quá trình tương tác với tác nhân. Nó phản ánh chuỗi các sự kiện diễn ra trong từng kịch bản ca sử dụng.

## **Câu 13. Trình bày sự khác nhau giữa một ca sử dụng và một kịch bản?**

* Kịch bản là những gì cần thiết
* Ca sử dụng là cách mà phần mềm sẽ thao tác với yêu cầu người dùng
* Kịch bản phải dễ đọc và hiểu đối với người dùng

Ca sử dụng sẽ mô tả một cách đầy đủ về cách phần mềm

## **Câu 14: Mục đích của hoạt động phân tích và thiết kế là gì ? Đầu vào và đầu ra cho hoạt động phân tích và thiết kế là gì ?**

- Hoạt động phân tích và thiết kế được thực hiện cho ba mục đích sau:

Thứ nhất, đây là hoạt động nhằm chuyển các yêu cầu thành một bản thiết kế hệ thống

Thứ hai, phân tích và thiết kế là hoạt động để hình thành kiến trúc chắn chắn cho hệ thống

Thứ ba, phân tích thiết kế là hoạt động nhằm đưa ra giải pháp thiết kế thích ứng với môi trường cài đặt và các yếu tố ràng buộc về trình diễn và thực thi của hệ thống.

* Đầu vào của hoạt động phân tích và thiết kế gồm có mô hình ca sử dụng, từ điển dữ liệu và đặc tả bổ sung.
* Đầu ra của hoạt động phân tích và thiết kế mô hình thiết kế là mô hình thiết kế

## **Câu 15: Kiến trúc phần mềm là gì? Trình bày ngắn gọn về kiến trúc khung nhìn “4+1”?**

* Kiến trúc phần mềm: là cấu trúc của các hệ thống đó bao gồm các phần tử phần mềm, các thuộc tính và mối quan hệ giữa chúng, ngoài ra cũng đề cập đến các tài liệu kiến trúc phần mềm của hệ thống, thuận tiện cho việc trao đổi thông tin giữa các thành viên trong dự án -> giúp quyết định ở mức cao và sử dụng các thành phần và mẫu thiết kế của dự án.

- kiến trúc khung hình 4+1 là một phương pháp để tái hiện kiến trúc phần mềm. Mô hình này dùng để tổ chức, phản ánh hệ thống từ 4+1 khung nhìn khác nhau, trong đó khung nhìn ca sử dụng địch hướng và chi phối các khung hình còn lại

## **Câu 16: Trình bày vai trò của pha phân tích? Nêu các bước trong Phân tích hướng đối tượng?**

* Các bước trong phân tích thiết kế hướng đối tượng:

+ Xây dựng biểu đồ use case: dựa trên tập yêu cầu ban đầu, người phân tích tiến hành xác định các tác nhân, use case và quan hệ giữa các use case để mô tả lại các chức năng của hệ thống. Một thành phần quan trọng trong biểu đồ UC là các kịch bản mô tả hoạt động của hệ thống trong mỗi UC cụ thể

+ Xây dựng biểu đồ lớp: xác định tên các lớp, các thuộc tính của lớp, một số phương thức và các quan hệ trong sơ đồ lớp

+ Xác định biểu đồ trạng thái: mô tả các trạng thái và chuyển tiếp trạng thái trong hoạt động của một đối tượng thuộc lớp nào đó

* Vai trò của pha phân tích:

+ Quyết định vòng đời của hệ thống

+ Cấu thành nên một hệ thống hoạt động hoạt động thực sự và định nghĩa một chiến lược cài đặt cho hệ thống

## **Câu 17: Mục đích của hoạt động phân tích kiến trúc là gì?**

* Xác định một kiến trúc tiêu biểu cho hệ thống dựa vào kinh nghiệm từ các hệ thống tương tự hay các vấn đề tương tự
* Xác định các mẫu kiến trúc, các cơ chế kiến trúc chính và các quy ước về mô hình hệ thống
* Xác định các chiến lược sử dụng lại

## **Câu 18: Cơ chế phân tích là gì? Cho ví dụ?**

* Cơ chế phân tích: Là một phần hay một giai đoạn trong quá trình phân tích để giảm độ phức tạp của phân tích và cải thiện tính nhất quán của nó bằng cách cung cấp cho các nhà thiết kế một biểu diễn ngắn gọn cho các hành vi phức tạp.
* Ví dụ: Bảo mật Security dùng cho các lớp cần bảo mật như: Người dùng, người quản lý, …

## **Câu 19: Mục đích của hoạt động phân tích ca sử dụng là gì?**

* Mô tả yêu cầu chức năng của hệ thống sau khi đã thỏa thuận giữa khách hàng và người phát triển
* Mô tả rõ rang và nhất quán hệ thống sẽ làm gì và được sử dụng trong suốt quá trình phát triển như thế nào
* Cung cấp cơ sở để kiểm tra thử nghiệm hệ thống

## **Câu 20: Lớp phân tích là gì? Nêu các kiểu mở rộng (stereotypes) của lớp phân tích? Trình bày vai trò và cách xác định các kiểu lớp: Lớp biên, lớp điều khiển, lớp thực thể?**

Lớp là mô tả thuộc tính, hành vi, ngữ nghĩa của một nhóm đối tượng, xác định thông tin nào sẽ được lưu trữ trong đối tượng và hành vi nào đối tượng có  
 Lớp biển: những lớp này có stereotype dạng <houndury>>  
 Lớp thực thể: những lớp này có stereotype dạng<<entity>>  
 Lớp điều khiển: những lớp này có stereotype dạng<<control>>

* Lớp biên: dành cho lớp nằm trên biên hệ thống, có thể là form, report, giao diện, … Khảo sát biểu đồ UC để tìm kiếm lớp biên
* Lớp thực thể: là lớp lưu trữ các thông tin và sẽ ghi vào bộ nhớ ngoài. Tìm chúng trong biểu đồ sự kiện và biểu đồ tương tác
* Lớp điều khiển: có trách nhiệm điều phối hoạt động của các lớp khác. Thông thường mỗi UC chỉ có 1 lớp điều khiển

## **Câu 21: Các biểu đồ tương tác được tạo ra như thế nào trong hoạt động phân tích ca sử dụng? Phân biệt biểu đồ tuần tự và biểu đồ cộng tác?**

* Để xây dựng biểu đồ tương tác ta bắt đầu từ luồng sự kiện

+ Xây dựng từng biểu đồ cho luồng chính, luồng thay thế, luồng lỗi

+ Sử dụng mẫu (pattern) xây dựng biểu đồ tương tác để giảm thời gian

* Biểu đồ trình tự:

+ Là biểu đồ theo thứ tự thời gian

+ Đọc biểu đồ từ đỉnh xuống đáy

+ Đọc biểu đồ bằng quan sát đối tượng và thông điệp

+ Mỗi đối tượng có vòng đời (lifeline)

* Biểu đồ cộng tác:

+ Tập trung vào quan hệ giữa các đối tượng, cấu trúc tổ chức của đối tượng, luồng dữ liệu trong kịch bản

+ Tương đối khó quan sát trình tự các thông điệp trong biểu đồ cộng tác

## **Câu 22: Phân biệt hai quan hệ include và extend trong biểu đồ ca sử dụng?**

* Extend được sử dụng khi mà có một UC được mở rộng từ 1 UC gốc nào đó. Nói cách khác UC A được thực hiện khi UC B hoàn thành thì UC A gọi là extend của UC B
* Include thường được sử dụng để mô tả việc phân nhỏ 1 UC thành nhiều bản sao nhỏ. Một UC được vẽ là include thì không thể được hoành thành khi mà các UC con của nó chưa được hoành thành

## **Câu 23: Phân biệt các dạng quan hệ trong biểu đồ lớp: Association, Generalization, Aggregation, Composition, Dependency?**

* Quan hệ kết hợp (**Association**):

+ Là quan hệ cấu trúc để mô tả tập liên kết

+ Khi đối tượng của lớp này gửi/nhận thông điệp đến từ đối tượng của lớp kia thì gọi là quan hệ kết hợp

+ Là kết nối ngữ nghĩa giữa các lớp

+ Aggregation là một dạng đặc biệt của Association

* Quan hệ thành phần (**Composition**):

+ Là quan hệ giữa tổng thể và bộ phận

+ Trong quan hệ này, 1 lớp biểu diễn cái lớn hơn còn lớp kia biểu diễn cái nhỏ hơn

+ Có 2 loại: tụ hợp và gộp

* Quan hệ tổng quát hóa (**Generalization**):

+ Là quan hệ kế thừa của hai phần tử như lớp, tác nhân, UC và gói

* Quan hệ phụ thuộc (**Dependency**)

+, Dependency là quan hệ chỉ ra một lớp tham chiếu đến một lớp khác

+, Khi thay đổi đặc tả lớp tham chiếu thì lớp sử dụng nó bị ảnh hưởng

## **Câu 24: Trình bày vai trò của pha thiết kế? Nêu các bước trong Thiết kế hướng đối tượng?**

\***Vai trò của pha thiết kế**:

* Trả lời câu hỏi “how” thay vì câu hỏi “what” như trong pha phân tích
* Xác định hệ thống sẽ được xây dựng như thế nào dựa trên kết quả của pha phân tích
* Đưa ra các phần tử hỗ trợ giúp cấu thành nên một hệ thống thông qua hoạt động thực sự
* Định nghĩa một chiến lược cài đặt cho hệ thống

**\*Các bước thiết kế hướng đối tượng**:

* Xây dựng các biểu đồ tương tác bao gồm biểu đồ tuần tự và biểu đồ cộng tác
* Xây dựng biểu đồ lớp chi tiết
* Thiết kế chi tiết
* Xây dựng biểu đồ thành phần và biểu đồ triển khai hệ thống
* Phát sinh mã, chuẩn bị cho cài đặt hệ thống

## **Câu 25: Mục đích của việc xác định các phần tử thiết kế là gì?**

* Phân tích kiến trúc định nghĩa các lớp của hệ thống. Tập trung vào lớp trên
* Phân tích use case: tập trung vào phân tích yêu cầu và phân bổ trách nhiệm cho lớp phân tích
* Xác định phần tử thiết kế:

+Tinh chỉnh các chi tiết của công việc kiến trúc

+ Các lớp phân tích được tinh chỉnh thành các phần tử thiết kế

## **Câu 26: Hệ thống con là gì ? Phân biệt hệ thống con và gói ? Trình bày mục đích sử dụng và cách xác định các hệ thống con ?**

- Hệ thống con bản chất nó cũng là một hệ thống nhưng là thành phần của một hệ thống khác. Những hệ thống mà chúng ta đang xem xét thực chất đều là các hệ thống con nằm trong một hệ thống khác và đồng thời cũng chứa các hệ thống con khác thực hiện những nhiệm vụ khác nhau của công việc.  
- Phân biệt gói và hệ thống con:

|  |  |
| --- | --- |
| Các gói :  Không cung cấp hành vi  Không đóng gói hoàn toàn nội dung của chúng  Không dễ bị thay thế | Các hệ thống con :  Cung cấp hành vi  Đóng gói hoàn toàn nội dung của chúng  Dễ bị thay thế |

- Các hệ thống con thường sử dụng để biểu diễn sản phẩm đã tồn tại hoặc các dịch vụ mà hệ thống sử dụng  
- Cách xác định các hệ thống con:

+ Xây dựng mô hình lớp thiết kế: bước này ta tập trung vào thiết kế các lớp thuộc cụm nghiệp vụ bao gồm các thực thể trong miền bài toán và các lớp hỗ trợ cần thiết khác

+ Xây dựng lược đồ cơ sở dữ liệu: quyết định cách lưu trữ dữ liệu và thiết kế khuôn dạng lưu trữ. Dữ liệu lưu trữ là dữ liệu không bị mất đi khi hệ thống ngưng hoạt động

+ Thiết kế giao diện người dùng: tiến hành xây dựng giao diện người dùng dựa trên phác thảo giao diện được đưa ra trong pha phân tích yêu cầu

## **Câu 27: Cơ chế phân tích trong thiết kế hướng đối tượng dùng để làm gì ? Tại sao cần phân loại các cơ chế phân tích ?**

+) Cơ chế phân tích trong thiết kế hướng đối tượng dùng để giảm độ phức tạp của phân tích và để cải tiến sự đồng nhất bằng cách cung cấp cho người thiết kế một sự biểu diễn ngắn gọn vè các hành vi phức tạp  
+) Cần phân loại các cơ chế phân tích vì mục đích của nó là để làm mịn các thông tin ban đầu các cơ chế phân tích xác định được.

## **Câu 28. Mục đích thiết kế ca sử dụng là gì ?**

- Nó giúp chúng ta hiểu yêu cầu, kiến trúc chức năng của hệ thống và chi phối tất cả các bản vẽ còn lại đồng thời để tạo ra một tài liệu mô tả tổng thể về hệ thống bằng cách xác định mục đích người dùng.

## **Câu 29. Mục đích của việc thiết kế lớp là gì? Các lớp được làm mịn trong thiết kế hướng đối tượng bằng cách nào ?**

- Mục đích: Thể hiện, đại diện cho sự trừu tượng của một hoặc một số lớp trong quá trình triển khai của hệ thống, đặc biệt nó cũng phải tương thích mới ngôn ngữ mà ta triển khai.

- Các lớp được làm mịn trong Thiết kế hướng đối tượng: Để làm mịn ta cần xác định các đối tượng tham gia, phân bổ trách nhiệm giữa các đối tượng, mô hình nên các thông điệp giữa các đối tượng, mô hình nên các mối quan hệ liên kết giữa các lớp.

## **Câu 30. Trình bày sự khác nhau giữa tiến trình (process) và luồng (thread), thao tác (operation) và phương thức (method) ?**

1. Sự khác nhau giữa tiến trình và luồng:

+Tất cả các luồng của một chương trình được chứa một cách hợp lí trong một tiến trình

+Luồng nhẹ hơn tiến trình

+Một tiến trình là một đơn vị thực thi bị cô lập trong khi luồng không bị cô lập và có chia sẻ bộ nhớ

+Một luồng không thể tồn tại riêng biệt, nó được gắn liền với một tiến trình. Mặt khác một tiến trình có thể tồn tại riêng lẻ

+Tại thời điểm hết hạn của một luồng, ngăn xếp liên kết của nó có thể được phục hồi vì mỗi luồng có ngăn xếp riêng. Ngược lại, nếu muột tiến trình chết, tất cả các luồng sẽ chết bao gồm cả tiến trình

2. Sự khác nhau giữa thao tác và phương thức:

+Operation là các quá trình mà một lớp cần để thực hiện hay thực hiện một hoạt động trên một đối tượng, còn phương thức thường chỉ là một hàm, một hoạt động phần thân của một hàm trong một đối tương.

+ Cả hai khác nhau khi bạn có tính đa hình. Nếu bạn có một lớp kiểu cha với ba kiểu con, mỗi kiểu sẽ ghi đè hoạt động "something" của kiểu cha, bạn có một Operation và bốn method triển khai nó.

**Biểu đồ tuần tự**







