**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 gắn gọn về quy trình phát triển RUP? Nêu ưu và nhược điểm của mô hình phát triển này?**

- Quy trình hợp nhất là một quy trình nghiệp vụ cho kỹ thuật phần mềm hướng đối tượng, dùng để mô tả một họ các tiến trình kỹ thuật phần mềm cùng chia sẻ cấu trúc và kiến trúc tiến trình chung.

- Ưu điểm:

+ Thường xuyên nhận phản hồi từ các cổ đông

+ Sử dụng các tài nguyên dự án 1 cách hiệu quả

+ 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

- Nhược điểm:

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

+ Quá trình phát triển có thể vượt qua tầm kiểm soát

+ Cần các chuyên gia để có thể đáp ứng được 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, Usecase, 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.

**5. Trình 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 usecase: Một mô hình Use case là một loại của ngôn ngữ mô hình hóa thống nhất(UML) được định nghĩa bởi Object Management Group(OMG). Mô hình Use case cung cấp cái nhìn trực quan đồ họa về nghiệp vụ của hệ thống(đại diện là Use case), người dùng hệ thống.

Mục đích: Mô hình hóa nhằm tạo ra một tài liệu mô tả tổng thể về hệ thống bằng cách xác định mục đích của người dùng.

+ Biểu đồ tương tác: Biểu đồ tương tác gồm hai loại biểu đồ, biểu đồ tuần tự và biểu đồ giao tiếp. Biểu đồ tương tác biểu diễn một tương tác, bao gồm tập các đối tượng, các quan hệ và các thông điệp giao tiếp giữa chúng. Biểu đồ tuần tự là biểu đồ tương tác chú trọng vào trình tự thời gian của các thông điệp, trong khi biểu đồ giao tiếp chú trọng vào tổ chức cấu trúc của các đối tượng trong quá trình truyền và nhận thông điệp. Biểu đồ tuần tự và biểu đồ giao tiếp có thẻ chuyển đổi tương đương với nhau.

+Biểu đồ lớp: Biểu đồ lớp biểu diễn một tập hợp các lớp, các giao diện, các cộng tác và các quan hệ giữa chúng. Đây là loại biểu đồ thông dụng nhất trong mô hình hóa các quan hệ thiết kế tĩnh của hệ thống. Biểu đồ lớp với các lớp tích cực cho phép biểu diễn khung nhìn tiến trình tĩnh của hệ thống.

+Biểu đồ đối tượng: Biểu đồ đối tượng biểu diễn một tập các đối tượng và các quan hệ giữa chúng. Biểu đồ đối tượng tái hiện hình chụp của hệ thống(snapshots), gôm các thể hiện của các lớp có trong các biểu đồ lớp. Giống như các biểu đồ lớp, biểu đồ đối tượng cũng cho phép biểu diễn khung nhìn thiết kế tĩnh hoặc khung nhìn tiến trình tĩnh của hệ thống. Tuy nhiên, đây là những góc nhìn phản ánh các hiện thực tron thế giới thực hoặc trong các bản mẫu.

+Biểu đồ trạng thái: Biểu đồ trạng thái biểu diễn một máy trạng thái, bao gốm các trạng thái, các chuyển, các sự kiện và các hoạt động. Biểu đồ trạng thái giải quyết khung nhìn động của hệ thống. Nó đặc biệt quan trọng trong việc mô hình hóa hành vi của một giao diện, lớp hay công tác và biểu diễn hành vi chú trọng vào trình tự sự kiện diễn ra với đối tượng. Kỹ thuật mô hình hóa này đặc biệt có ích trong trong việc mô hình hóa các hệ thống phản ứng với các sự kiện (reactive systems).

+Biểu đồ hoạt động: Biểu đồ hoạt động là trường hợp đặc biệt của biểu đồ trạng thái, diễn đật luồng hoạt động trong hệ thống. Biểu đồ hoạt động cho phép mô tả khung nhìn động của hệ thống. Biểu đồ hoạt động cho phép mô tả khung nhìn động của hệ thống. Chúng đặc biệt quan trọng trong mô hình hóa chức năng của hệ thống và nhấn mạnh luồng điều khiển giữa các đối tượng.

+Biểu đồ thành phần: Biểu đồ thành phần biểu diễn các tổ chức các thành phần và sự phụ thuộc giữa chúng. Biểu đồ thành phần cho phép mô tả khung nhìn cài đặt tĩnh của hệ thống. Chúng liên quan đến các biểu đồ lớp theo cashc một thành phần có thể tương ứng với một hay nhiều lớp, giao diện, hoặc cộng tác.

+Biểu đồ triển khai: Biểu đồ triển khai biểu diễn cấu hình của các nút xử lý và các thành phần thực thi trên các nút xử lý đó. Biểu đồ triển khai cho phép mô tả khung nhìn triển khai tĩnh của một kiến trúc. Chúng liên quan đến các biểu đồ thành phần theo cách một nút xử lý được liên kết với một hay nhiều thành phần.

**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 usecase, khung nhìn máy trạng thái, khung nhìn hoạt động, khung nhìn tương tác, khung nhìn triển khai, khung nhìn quản lý mô hình ?**

-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 tĩnh phản ánh cấu trúc đối tượng của hệ thống. Cấu trúc dữ liệu và các đặc điểm hành vi được thống nhất trong cấu trúc đối tượng. Khung nhìn tĩnh mô tả các khai báo hành vi dưới dạng khai báo các thao tác. Nó không mô rả chi tiết về ành vi dưới dạng khai báo các thao tác. Nó không mô tả chi tiết về hành vi động như các khung nhìn tương tác hay khung nhìn máy trạng thái. Phần tử chính trong khung nhìn tĩnh là các phân lớp ( chẳng hạn như lớp, giao diện và kiểu dữ liệu) và quan hệ giữa chúng ( kết hợp, tổng quát hóa và quan hệ phụ thuộc). Trong khung nhìn này, mỗi đối tượng và một đơn vị cơ bản làm nên hệ thống và gói là đơn vị tổ chức chung để quản lý các nội dung của mô hình.

-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 thống 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.

**7. Nêu một số vấn đề trong phát triển phần mềm hướng đối tượng ? Trình bàynội 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

+Yêu cầu không được đáp ứng

+Yêu cầu thay đổi quá nhanh

+Các mô đun không khớp

+Bảo trì khó

+Phát HIện lỗi muộn

+Chất lượng kém

+Trình Diễn kém

+Xung đột với người phát triển

+Nhập nhằng giữa bản xây dựng và bản chuyển giao

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

Bộ kinh nghiệm thực tiễn ( “Best Practices”) là tập hợp các phương pháp phát triển phần mềm đã được kiểm nghiệm bằng các phần mềm thương mại. Tính đúng đắn của chúng đã được khẳng định thông qua quá trình được sử dụng thường xuyên và thành công trong công nghiệp và các tổ chức. Bộ kinh nghiệm thu được từ hàng ngàn khách hàng thực hiện trên hàng ngàn dự án và từ các chuyên gia phần mềm.

**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

**THIẾU**

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

- Mô hình miền là một đại diện trực quan của các lớp khái niệm, các đối tượng và mối quan hệ giữa chúng trong phạm vi bài toán

- Mô hình miền là thành phần quan trọng nhất được tạo ra trong giai đoạn phân tích hướng đối tượng, là cơ sở cho việc thiết kế phần mềm.

- Trong UML, mô hình miền được biểu diễn bằng một tập hợp các sơ đồ lớp, trong đó chỉ hiển thị:

+ Đối tượng hoặc các lớp khái niệm

+ Các mối quan hệ giữa các lớp khái niệm

+ Các thuộc tính của các lớp khái niệm

- Mô hình miền xác định cách mà các yếu tố bên trong của hệ thống tương tác với nhau ( hành vi bên trong ) để thực hiện các hành vi bên ngoài.

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

- Đặc tả bổ sung:

+ Tài liệu đặc tả bổ sung bao gồm các yêu cầu bổ sung quan trọng cho mô hình usecase, những yêu cầu không được đề cập trong mô hình usecase. Tài liệu này cùng với tài liệu usecase sẽ cung cấp thông tin đầy đủ cho một đặc tả yêu cầu về hệ thống.

+ Các yêu cầu chức năng và phi chức năng trong tài liệu đặc tả bổ sung thường là các ràng buộc cho việc thực thi hệ thống. Các ràng buộc có thể được phân lớp như sau:

\* Ràng buộc chức năng: gồm các yêu cầu chức năng được dùng chung và tham chiếu đến từ các usecase khác nhau.

\* Tính có thể sử dụng được: các yêu cầu liên quan đến khả năng có thể sử dụng được của hệ thống.

\* Tính có thể tin cậy được: gồm bất cứ yêu cầu nào liên quan đến tính tin cậy của hệ thống.

\* Tính trình diễn: gồm các ràng buộc về đặc điểm thực thi của hệ thống như các yêu cầu về thời gian phản hồi của hệ thống.

\* Các ràng buộc thiết kế: gồm bất kỳ yêu cầu, ràng buộc về hệ thống đang được xây dựng.

+ Các đặc tả bổ sung đi kèm với mô hình usecase. Chúng được tạo và xem xét từ rất sớm trong pha xây, góp phần làm rõ phạm vi của hệ thống. Các đặc tả này sẽ được chi tiết hóa trong các lần lặp sau trong suốt pha chi tiết và pha xây dựng.

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

- Từ điển thuật ngữ bao gồm các thuật ngữ trong miền của hệ thống, được mô tả ở dạng văn bản và được dùng chung cho tất cả các mô hình của hệ thống. Nó thường được phát biểu định nghĩa các thuật ngữ quan trọng được sử dụng trong dự án.

- 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

**12. Mục đích của việc xác định yêu cầu là gì ? Nêu vai trò của biểu đồ usecase 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 usecase mô tả những gì mà hệ thống sẽ làm.

+ Mô hình usecase 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 cầu của khách hàng và người dùng.

+ Mô hình usecase bao gồm các usecase và các tác nhân. Mỗi usecase 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 usecase.

**13.Trình bày sự khác nhau giữa một usecase và một kịch bản ?**

Một usecase liên quan đến một tác nhân và dòng chảy mà một tác nhân cụ thể thực hiện trong một chức năng hoặc đường dẫn cụ thể. Chúng thường được nhóm lại để bạn có một "tập hợp" các **trường hợp sử dụng** để tính đến từng **kịch bản** .

Một **kịch bản** liên quan đến một tình huống mà có thể có diễn viên duy nhất hoặc nhiều mà phải mất một chức năng nhất định hoặc đường dẫn đến giải quyết **tình huống** .

Bạn có thể thấy sự khác biệt chính là " **phối cảnh** " ở đây. Thông thường**, trường hợp sử dụng**là chi tiết hơn so với **kịch bản** . Trong sự phát triển của những điều này, nó thường liên quan đến việc đưa ra một kịch bản và sau đó xác định tất cả các **trường hợp sử dụng** phù hợp với**kịch bản** cụ thể đó . Vì các **trường hợp sử dụng**ghi lại **kịch bản** sau khi viết dấu chân tài liệu thường chỉ là một hoặc không phải cả hai, đó cũng có thể là lý do tại sao tên đôi khi được thay thế cho nhau thay vì đứng như những thứ riêng biệt. Nhiều dự án không yêu cầu mức độ chi tiết của usecase và chỉ tuân theo các **kịch bản** thay vì phá vỡ nó thành các luồng diễn viên cụ thể trong **các trường hợp sử dụng** .

**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 usecase, 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

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

- Kiến trúc phần mềm chứa đựng một tập các quyết định thiết yếu về tổ chức hệ thống phần mềm. nó phản ánh các khía cạnh cấu trúc(khía cạnh tĩnh), hành vi (khía cạnh động)và cách tổ chức phân cấp trong hệ thống.Về khía cạnh cấu trúc, kiến trúc phần mềm cho phép lựa chọn các phần tử cấu trúc và các giao diện tương tác giữa chúng để tạo hình nên hệ thống .Hành vi của hệ thống được mô tả thông qua các động tác giữa phần tử cấu trúc đó . các phần tử cấu trúc và hành vi này có thể được nhóm gộp thành các hệ thống con của hệ thống. thông thường cách tổ chức hệ thống đó được thực hiện theo kinh nghiệm thông qua các mâu thuẫn và kiểu kiến trúc có sẵn . Tóm lại, kiến trúc phần mềm được diễ đạt ở dạng biểu thức sau: kiến trúc = các phần tử + các khuôn dạng + cơ sở hợp lý

- 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 usecase 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 ?**

Việc phân tích một hệ thống bao gồm các nhiệm vụ: Thiết lập một cách nhìn tổng quan rõ ràng về hệ thống và các mục đích chính của hệ thống cần xây dựng; Liệt kê các nhiệm vụ mà hệ thống cần thực hiện; Phát triển một bộ từ vựng để mô tả bài toán cũng như những vấn đề liên quan và đưa ra hướng giải quyết bài toán. Việc thiết kế một hệ thống, thực chất là: Xác định hệ thống sẽ được xây dựng như thế nào dựa trên kết quả của việc phân tích; Đưa ra các phần tử hỗ trợ giúp cấu thành nên một hệ thống hoạt động thực sự và định nghĩa một chiến lược cài đặt cho hệ thống .

Giai đoạn phân tích thiết kế hệ thống là giai đoạn chuyển những yêu cầu từ khách hàng thành phần mềm máy tính. Đây là giai đoạn quan trọng nhất trong tiến trình phát triển phần mềm vì giai đoạn này quyết định vòng đời của hệ thống. Trong các hệ thống thông tin vừa và nhỏ, một phân tích viên có thể là người lập trình cho hệ thống. Tuy nhiên, đối với một hệ thống lớn thì bộ phận phân tích viên phải là một tập thể và có đủ khả năng nắm bắt các lĩnh vực hoạt động của những yêu cầu được đặt ra từ khách hàng. Đồng thời, ở giai đoạn này việc chọn lựa phương pháp và mô hình phân tích thiết kế cũng có ảnh hưởng không nhỏ đến cấu trúc và tốc độ truy xuất dữ liệu của hệ thống.

Mặt khác, ngoài việc đáp ứng các yêu cầu nghiệp vụ của người dùng một phần mềm tốt cần có khả năng thích ứng và mở rộng. Vì vậy, phần mềm cần được phân tích thiết kế tốt để đứng vững trước những biến đổi của môi trường, dù từ phía cộng đồng người dùng hay từ phía công nghệ vẫn có thể dễ dàng phát triển theo yêu cầu của người dùng mà không cần phải sửa chữa nhiều.

Các bước trong phân tích hướng đôi tượng là:

1. Xác định các đối tượng và nhóm thành các lớp.

2. Xác định mối quan hệ giữa các lớp.

3. Tạo sơ đồ mô hình đối tượng người dùng.

4. Xác định thuộc tính đối tượng người dùng.

5. Xác định các hoạt động nên được thực hiện trên các lớp.

6.Đánh giá thuật ngữ

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

Phân tích kiến trúc là phân tích tất cả các yếu tố tạo thành đồ án kiến trúc, giúp người xem dễ hiểu đồ án của bạn, hiểu tư duy logic của bạn.

mục đích của phân tích kiến trúc là giúp người xem dễ hiểu đồ án của bạn, Một đồ án mà không ai hiểu ý nghĩa của nó, không có lập luận logic tạo nên nó, chỉ đẹp long lanh thôi chưa chắc đã thuyết phục khách hàng bỏ tiền để xây dựng, do đó chúng ta cần diễn giải phân tích ra để người xem dễ hiểu, tiết kiệm thời gian để hiểu đồ án.

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

Phân tích:

Là giai đoạn trung tâm khi xây dựng 1 hệ thống thông tin, giai đoạn này bao gồm các giai đoạn và khởi sự ngay trong giai đoạn lập kế hoạch. Phân tích bao gồm các công đoạn sau:

Phân tích hiện trạng:

Giai đoạn này nhằm hiểu rõ tình trạng hoạt động của hệ thống cũ trong mục đích hoạt động của tổ chức. Cụ thể, nó bao gồm các công việc:

- *Tìm hiểu hiện trạng*: thông qua việc nghiên cứu hồ sơ, tài liệu để tìm hiểu thông tin chung về ngành dọc của tổ chức.

* *Tìm hiểu hoạt động hiện tại của tổ chức*
* Xác định các thành phần tham gia trong tổ chức
* Các nhiệm vụ của các tổ chức thành viên và các tổ chức bên ngoài có liên quan
* Các mối quan hệ thông tin giữa các thành viên trong tổ chức

Phân tích khả thi và lập hồ sơ nhiệm vụ:

Phân tích khả thi phải tiến hành trên 3 mặt:

. *Phân tích khả thi về kỹ thuật*: xem xét khả năng kỹ thuật hiện có để đề xuất giải pháp kỹ thuật áp dụng cho hệ thống thông tin mới.

. *Phân tích khả thi kinh tế*: xem xét khả năng tài chính để chi trả cho việc xây dựng hệ thống thông tin mới cũng như chỉ ra những lợi ích mà hệ thống sẽ đem lại.

. *Phân tích khả thi hoạt động*: khả năng vận hành hệ thống trong điều kiện khuôn khổ, điều kiện tổ chức và quản lý cho phép của tổ chức.

Sau đó, người phân tích phải định ra một vài giải pháp và so sánh, cân nhắc các điểm tốt và không tốt của từng giải pháp. Tóm lại, trong giai đoạn này người phân tích phải tìm *ra một điểm cân bằng giữa nhu cầu và khả năng*.

Sau khi đã chọn lựa xong giải pháp người phân tích cần phải *lập hồ sơ nhiệm vụ*. Công việc này nhằm mục đích:

* Định hình các chức năng hệ thống cần đạt được.
* Định ra các thủ tục xây dựng quan niệm và thực hiện hệ thống.
* Định hình sơ lược giao diện của hệ thống với người sử dụng trong tương lai. Làm các bản mẫu (prototype) để NSD hình dung được hệ thống trong tương lai.

Tóm lại, lập hồ sơ nhiệm vụ là một thỏa thuận không chính thức giữa 3 phía: Người phân tích, Chủ đầu tư và Người sử dụng.

Xây dựng mô hình hệ thống chức năng:

Người phân tích dựa vào kết quả phân tích để xây dựng mô hình nghiệp vụ của hệ thống, từ đó làm rõ mô hình thông tin và mô hình họat động của hệ thống. Trong toàn bộ hoạt động phân tích thì đây là giai đoạn quan trọng nhất. Quá trình tìm hiểu và xây dựng mô hình hệ thống được gọi là hoàn tất nếu không còn một phản hồi nào từ phía chủ đầu tư.

Thiết kế:

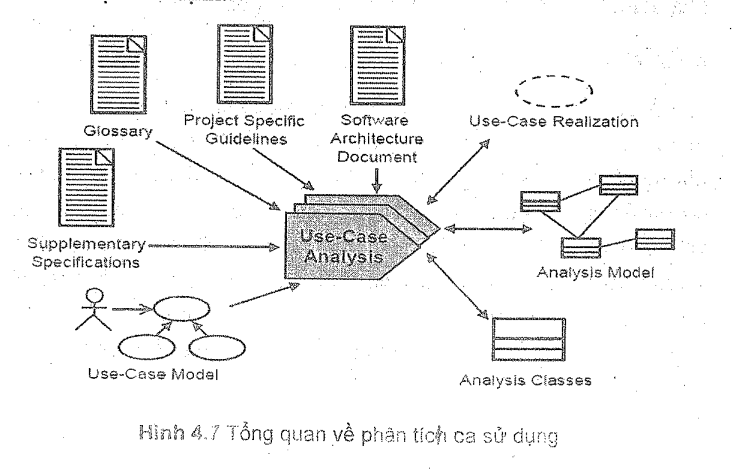
Thiết kế và phân tích không phải là hai giai đoạn rời nhau. Thiết kế hệ thống sẽ cho một phương án tổng thể hay một mô hình đầy đủ của hệ thống thông tin. Nó bao gồm tất cả các đặc tả về hình thức và cấu trúc của hệ thống. Trong giai đoạn thiết kế người phân tích phải xác định một cách chi tiết:

* Các thông tin.
* Các qui tắc phát sinh, tiếp nhận và xử lý thông tin
* Các kiểu khai thác
* Các phương tiện cứng và mềm được sử dụng trong hệ thống.

Ví dụ:Ví dụ hệ thông tin quản lý, hệ hỗ trợ quyết định, hệ quản lý bán hàng cần biết hàng tồn kho, ngân sách hàng năm để lập kế hoạch sản xuất, phân tích chi phí,

**19. Mục đích của hoạt động phân tích usecase là gì?**

* Để xác định các lớp phân tích thực thi các luồng sự kiện của usecase.
* Để phân bố các hành vi usecase cho các lớp này sử dụng các hiện thực hóa các usecase.
* Để xác định các trách nhiệm, các thuộc tính và các liên kết
* của các lớp phân tích.
* Để liên hệ các lớp phân tích đến từng cơ chế kiến trúc đã được xác định.



Và như minh họa, đầu vào cho hoạt động phản tích usecase gồm các chế tác sau: Từ điển thuật ngữ, Các đặc tả bổ sung. Usecase, Mô hình usecase, Hiện thực hóa usecase. Tài liệu kiến trúc phần mềm, Các lớp phân tích, Mô hình phân tích và Các hướng dẫn đặc thù cho dự án. Đầu ra của hoạt động phân tích usecase gồm các chế tác sau: Các lớp phân tích, Mô hình phân tích, Các hiện thực hóa usecase. Lưu ý, đối với một số hệ thống không quá phức tạp thì chúng ta có thể bỏ qua việc xây dựng mô hình phân tích một cách độc lập.

Hoạt động phân tích usecase thường trải qua các bước chính sau. Thứ nhất, chúng phải thẩm định và sửa đổi các mô tả usecase đã được xác định trong khâu nắm bắt yêu cầu. Các thay đổi ở đây chủ yếu là bổ sung các thông tin để chi tiết hóa các mô tả usecase, Hàm cơ sở cho việc xây dựng mô hình. Tiếp đó, chúng ta khảo sát luồng kịch bản usecase, xác định các lớp phân tích và phân bổ các trách nhiệm usecase cho các lớp phân tích. Dựa vào sự phân bố này cũng với các cộng tác lớp phân tích, chúng ta bắt đầu mô hình hóa mối quan hệ giữa các lớp phân tích. Một khi các usecase đã được phân tích, chúng ta tiến hành tổng hợp kết quả phân tích, đảm bảo các lớp được làm tài liệu. Tiếp đó, chúng ta cần phải đảm bảo tính nhất quán trong mô hình phân tích.

**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 phân tích là những lớp đầu tiên (proto-classes) chứa đựng các hành vi thiết yếu tương ứng với các trách nhiệm được gần. Chúng là những hình dung ban đầu về sự hợp thành của hệ thống. Các lớp phân tích hiếm khi được giữ nguyên trong khẩu thực thi.

Chúng ta tập trung vào kỹ thuật tìm kiếm lớp phân tích dựa vào các khía cạnh khác nhau của hệ thống. Đó là những khía cạnh sau:

* Biên giữa hệ thống và các tác nhân
* Thông tin hệ thống sử dụng
* Điều khiển logic trong hệ thống

Tương ứng với ba khía cạnh của hệ thống làm cơ sở phân tách hệ thống theo tiêu chí về tính thay đổi như trình bày ở trên, chúng ta xác định được các kiểu của các lớp phân tích sau

* Lớp biển: những lớp này có stereotype dụng <houndury>>
* Lớp thực thể: những lớp này có stereotype dạng<<entity>>
* Lớp điều khiển: những lớp này có stereotype dụng<<control>
* *Xác định lớp biên:*

Lớp biển đóng vai trò là cầu nối giữa giao diện và những thứ bên ngoài hệ thống, thuộc về môi trường. Lớp biên làm tách biệt hệ thống khỏi các thay đổi liên quan đến môi trường (những thay đổi về giao diện hệ thống cũng như thay đổi về yêu cầu người dùng). giữ cho những thay đổi này không ảnh hưởng đến phần còn lại của hệ thống. Một hệ thống có một vài kiểu lớp biến sau:

* Lớp giao diện người dung: Là cầu nối giao tiêp giữa hệ thống và người dùng.
* Lớp giao diện hệ thống: Là cầu nối giao tiếp đến các hệ thống khác. Lớp biên giao tiếp với 1 lớp bên ngoài có trách nhiệm quản lý sự hội thoại với hệ thống bên ngoài, cung cấp cho hệ thống đang phát triển giao diện đến hẹ thống ngoài đó.
* Lớp giao diện thiết bị: Chúng cung cấp giao diện cho các thiết bị nơi mà các sự kiện bên ngoài được tiếp nhận. các lớp biên này đảm nhiệm trách nhiệm của các thiết bị và các cảm ứng.
* *Xác định lớp thực thể:*

Đối tượng thực thể tái hiện các khái niệm chính được phát triển của hệ thống. Lớp thực thể cung cấp một góc nhìn khác để hiểu về hệ thống. Các lớp thực thể và mối quan hệ giữa chúng làm nên cấu trúc dữ liệu logic của hệ thống. Nắm được cấu trúc dữ liệu có thể giúp chúng ta hình dung được những gì mà hệ thống cung cấp cho người dùng. Sau đây là một số nguồn thông tin chính để xác định các lớp thực thể sau:

* Từ diễn thuật ngữ (được xác định trong khâu nằm bất yêu cầu)
* Mô hình miền nghiệp vụ (được phát triển trong khâu mô hình hỏa nghiệp vụ)
* Mô tả luồng sự kiện usecase (được phát triển trong khâu nằm bắt yêu cầu)
* Các trừu tượng chính (được xác định trong khẩu phần tích kiến trúc)

Như đã đề cập trước đây, đôi lúc chúng ta cần phải mô hình hóa thông tin về tác nhân của hệ thống. Lưu ý đây không phải là việc mô hình hóa tắc nhân, nhân tố phía ngoài hệ thống. Trong trường hợp này thông tin về tác nhân sẽ được mô hình hóa bởi một đối tượng thực thể.

Lớp thực thể đóng vai trò là các kho lưu thông tin của hệ thống. Chúng thường được dùng để tái hiện các khái niệm chính mà hệ thống quản lý. Các đối tượng thực thể (thể hiện của các lớp thực thể) được sử dụng để lưu trữ và cập nhật thông tin về các hiện tượng. chẳng hạn như các sự kiện hay các đối tượng trong thế giới thực. Lớp thực thể bao gồm các thuộc tính và các mối quan hệ thường. phải được lưu trữ bền vững trong suốt vòng đời của hệ thống.

* *Xác định lớp điều khiển:*

Các lớp điều khiển đóng vai trò điều phối hành vi trong hệ thống. Một hệ thống có thể thực hiện theo các usecase mà không cần các lớp thực thể, nghĩa là chỉ với các lớp thực thể và lớp biên. Điều này là hoàn toàn đúng cho các usecase liên quan đến chỉ những thao tác về lưu trữ thông tin. Đối với các usecase phức tịp hơn, nhìn chung hệ thống phải cần đến một hay nhiều lớp điều khiển để điều phối hành vi các đối tượng khác trong hệ thống. Lớp điều khiển có thể dăng vải trò cho việc quản lý giao dịch, điều phối thì nguyên hay điều khiển lỗi.

Các lớp điều khiển phân tách một cách hiệu quả giữa các đối tượng biển và đối tượng thực thể, làm cho hệ thống dễ thích ứng với các thay đổi từ phía biên của hệ thống. Chủng có thể phân tách các hành vì chuyên biệt trong cả sử dụng khỏi các đối tượng thực thể, giúp cho các đối tượng đó dễ được sử dụng chung cho các cả sử dụng hay các hệ thống khác. Hành vi mà các đối tượng điều khiển cung cấp thường mang những đặc điểm sau:

* Những hành vi đó thường độc lập với bên ngoài, nghĩa là không thay đổi khi bên ngoài môi trường hệ thống thay đổi. Chủng xác định logic điều khiển và tổ chức các giao dịch trong usecase.
* Đó là những hành vi ít bị thay đổi khi cấu trúc và hành vi bên trong của các lớp thực thể thay đổi. Và thậm chí hầu như không bị thay đổi nếu các lớp thực thể được tổ chức bao gối tốt.
* Đô là những hành vi sử dụng hay thiết lập nội dung của các lớp thực thể, và do đó chủng cần phải phối hợp với các hành vị của các lớp thực thể đó.
* Hành vi của các lớp điều khiển thưởng không được thực thi theo một cách duy nhất mỗi khi chúng được kích hoạt. Tùy vào tình huống, kịch bản usecase mà chúng thực thi theo các cách khác nhau.

**21.Các biểu đồ tương tác được tạo ra như thế nào trong hoạt động phân tích usecase ? Phân biệt biểu đồ tuần tự và biểu đồ cộng tác ?**

Các đối tượng tương tác với nhau để thực hiện một kịch bản. Nhìn chung, chúng ta có thể mô tả những tương tác như theo hai cách, hoặc tập trung vào từng đối tượng riêng là hoặc tập trung vào khung cảnh mà trong đó một nhóm các đối tượng tương tác với nhau. Lưu ý rằng cả hai cách được sử dụng bổ trợ cho nhau. Với cách thứ nhất, điển hình là tiếp cận sử dụng các máy trong thái, nó cung cấp một khung nhìn chi tiết về hành vi được phản ánh qua từng đối tượng riêng là. Các đặc tả đó thưởng đủ chính xác để chuyển trực tiếp sang mà thực thi. Tuy nhiên, đặc là đó không giúp ta nắm bắt được chức năng tổng thể của hệ thống, khi mà máy trọng thái tại một thời điểm, chỉ tập trung vào một đối tượng. Để có thể xác định được hành vi tổng thể của hệ thống, chúng ta cần phải kết hợp các hiệu ứng của nhiều máy trạng thái với nhau. Ngược lại khung nhìn tương tác trong tiếp cận thứ hai sẽ cung cấp một khung nhìn tổng thể về hành vi của một tập các đối tượng. Khung nhìn này được mô hình hóa bởi các tương tác trên các phân lớp có cấu trúc (ttructured classifiers) và các cộng tác (collaborations).

* Phân biệt biểu đồ tuần tự với biểu đồ cộng tác:
  + Biểu đồ tuần tự:
    - Biểu đồ tuần tự: Biểu diễn mối quan hệ giữa các đối tượng, giữa các đối tượng và tác nhân theo thứ tự thời gian. Biểu đồ tuần tự nhấn mạnh thứ tự thực hiện của các tương tác.
  + Biểu đồ cộng tác:
    - Biểu đồ cộng tác: Là biểu đồ tương tác biểu diễn mối quan hệ giữa các đối tượng; giữa các đối tượng và tác nhân nhấn mạnh đến vai trò của các đối tượng trong tương tác.
    - Biểu đồ cộng tác cũng có các message với nội dung tương tự như trong biểu đồ tuần tự. Tuy nhiên, các đối tượng được đặt một cách tự do trong không gian của biểu đồ và không có đường life line cho mỗi đối tượng. Các message được đánh số thể hiện thứ tự thời gian.

**22. Phân biệt hai quan hệ include và extend trong biểu đồ Use Case?**

– Quan hệ <<Include>>: use case này sử dụng lại chức năng của use case kia.

– Quan hệ <<Extend>>: use case này mở rộng từ use case kia bằng cách thêm vào một chức năng cụ thể

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

Giữa các lớp có các dạng quan hệ cơ bản như sau:

– Quan hệ kết hợp (Association): là một sự nối kết giữa các lớp, cũng có nghĩa là sự nối kết giữa các đối tượng của các lớp này.

– Khái quát hóa (Generalization): Khái quát hóa là mối quan hệ giữa một lớp có các đặc trưng mang tính khái quát cao hơn và một lớp có tính chất đặc biệt hơn. Trong sơ đồ lớp, mối quan hệ khái quát hóa chính là sự kế thừa của một lớp từ lớp khác. Quan hệ khái quát hoá được biểu diễn bằng một mũi tên có tam giác rỗng gắn ở đầu

– Quan hệ cộng hợp (Aggregation): là dạng quan hệ mô tả một lớp A là một phần của lớp B và lớp A có thể tồn tại độc lập. Quan hệ cộng hợp được biểu diễn bằng một mũi tên gắn hình thoi rỗng ở đầu hướng về lớp bao hàm.

– Quan hệ gộp (Composition): Một quan hệ gộp biểu diễn một quan hệ kiểu tổng thể-bộ phận. Lớp A có quan hệ gộp với lớp B nếu lớp A là một phần của lớp B và sự tồn tại của đối tượng lớp B điều khiển sự tồn tại của đối tượng lớp A. Quan hệ này được biểu diễn bởi một mũi tên gắn hình thoi đặc ở đầu.

– Quan hệ phụ thuộc (Dependency): Phụ thuộc là mối quan hệ giữa hai lớp đối tượng: một lớp đối tượng A có tính độc lập và một lớp đối tượng B phụ thuộc vào A; một sự thay đổi của A sẽ ảnh hưởng đến lớp phụ thuộc B.

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

Trong tiến trình phát triển phần mềm nói chung, bước thiết kế hướng đối tượng có vai trò như sau:

- Mục tiêu của pha thiết kế là phải 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 hoạt động thực sự.

- Định nghĩa một chiến lược cài đặt cho hệ thống. Các đặc trưng của pha thiết kế hướng đối tượng bao gồm:

- Mô hình hóa chi tiết hệ thống dựa trên các lớp, các đối tượng trong miền ứng dụng của hệ thống đó.

- Thiết kế dựa trên chiến lượng trừu tượng hoá phân cấp dữ liệu (hierachical data abstraction) trong đó các thành phần sẽ được thiết kế từ các lớp, đối tượng, các module và các tiến trình.

- Các phương thức thường được thiết kế trong mối quan hệ với các đối tượng xác định hoặc một lớp các đối tượng đó.

Các bước thiết kế hướng đối tượng

Dựa trên các kết quả của pha phân tích, pha thiết kế hướng đối tượng được chia thành các bước như sau:

+ 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: thực hiện hoàn chỉnh sơ đồ lớp, xác định và biểu diễn đầy đủ các phương thức cho từng lớp, xác định mối quan hệ giữa các lớp.

+ Thiết kế chi tiết: xây dựng các biểu đồ động cho các phương thức phức tạp trong các lớp và xây dựng bảng thiết kế chi tiết cũng như kế hoạch cài đặt và tích hợp.

+ Xây dựng biểu đồ thành phần và biểu đồ triển khai hệ thống

+ Phát sinh mã, chuẩn bị cho cài đặt hệ thống

**25. Mục đích của việc xác định các phần tử thiết kế là gì ?**Mục đích của việc xác định các phần tử thiết kế là phân tích sự tương tác giữa các lớp phân tích để xác định các phần tử của Mô hình thiết kế.

**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 (subsystem) là một phần tử mô hình có ngữ nghĩa giống với một gói (có thể chứa các phần tử mô hình khác) và một lớp (có hành vi).   
Một hệ thống con thực thi một hoặc nhiều giao diện -> định nghĩa hành vi mà hệ thống con đó có thể thực hiện  
Một hệ thống con có thể được biểu diễn như một gói UML, stereotype dạng <<subsystem>>  
+ 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 :   
Khi lớp phân tích quá phức tạp:   
 Các hành vi của lớp phân tích không thể chỉ là trách nhiệm của một mình nó  
 Các trách nhiệm của nó cần phải được sử dụng lại.  
 => Lớp phân tích đó cần phải được làm mịn thành một hệ thống con.

**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.

**28. Mục đích thiết kế usecase là gì ?**

- Mục đích của usecase là để xác định một mô đun hành vi mà không cần biết cấu trúc bên trong của chủ thể. Định nghĩa của usecase phải bao gồm tất cả hành vi nó đáp ứng.

**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 của thiết kế lớp là để đảm bảo các lớp cung cấp các hành vi mà Hiện thực hóa usecase yêu cầu, đảm bảo cung cấp các thông tin đầy đủ để cài đặt lớp. Ngoài ra hoạt động này cũng để nắm bắt các yêu cầu phi chức năng đến các lớp và tích hợp các cơ chế thiết kế được sử dụng bởi các lớp.

- Xác định các phần tử thiết kế là một hoạt động thuộc giai đoạn làm mịn thiết kế hệ thống (Refine the Architecture) trong giai đonạ Phân tích usecase. Chính trong quá trình xác định các phần tử thiết kế, chúng ta quyết định lớp phân tích nào thực sự là một lớp, lớp nào chuyển sang hệ thống con, và lớp nào là các thành phần hệ thống không cần thiết kế.

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

- Sự khác nhau giữa tiến trình và luồng là không gian bộ nhớ trong đó nó thực hiện:

+ Một tiến trình thực hiện trong không gian bộ nhớ, được đóng gói và bảo vệ cấu trúc nội tại. Một tiến trình có thể được xem như một hệ thống của chính nó, được khởi tạo bởi một chương trình thực hiện được. Một tiến trình có thể chứa nhiều luồng.

+ Một luồng thực hiện trong một không gian bộ nhớ ở đó nó có thể chia sẻ với các luồng khác.

- Sự khác nhau giữa thao tác (operation) và phương thức (method):

+ Thao tác cần được đặt tên theo giá trị trả về của chúng, cách tốt nhất đặc tả các thao tác và các tham số của chúng là sử dụng cú pháp và ngữ nghĩa của ngôn ngữ cài đặt.

+ Một phương thức đặc tả sự cài đặt của một thao tác, Nó mô tả làm thế nào (HOW) thao tác thực hiện, không phải mô tả phải làm gì (WHAT), khi mô tả phương thức chúng ta cần mô tả làm thế nào để các thao tác được cài đặt, làm thế nào các thuộc tính được cài đặt và sử dụng để cài đặt các thao tác.