PHÂN TÍCH THIẾT KẾ HTTT

1. Hệ thống thông tin là gì? Có những loại hệ thống thông tin nào?

* Hệ thống thông tin là một hệ thống bao gồm các yếu tố có quan hệ với nhau cùng làm nhiệm vụ thu thập, xử lý, lưu trữ, phân phối thông tin, dữ liệu và cung cấp một cơ chế phản hồi để đạt được một mục tiêu định trước.
* Hệ thống thông tin đã được ứng dụng trong mọi lĩnh vực khác nhau của đời sống xã hội. Tùy theo quan điểm mà có thể phân loại các hệ thống thông tin theo các tiêu chí khác nhau. Xét về mặt ứng dụng, hệ thống thông tin có thể chia thành một số dạng như sau:
* Hệ thống thông tin quản lý: bao gồm các hệ thống thông tin hỗ trợ các hoạt động nghiệp vụ và quản lý của các doanh nghiệp, các tổ chức. Vd các hệ thống quản lý nhân sự, hệ thống kế toán, hệ thống tính cước và chăm sóc khách hàng, hệ thống quản lý thư viện, hệ thống đào tạo trực tuyến,….
* Các hệ thống Website: là các hệ thống có nhiệm vụ cung cấp thông tin cho người dùng trên môi trường mạng internet. Các hệ thống website có đặc điểm là thông tin cung cấp cho người dùng có tính đa dạng (có thể là tin tức hoặc các dạng file đa phương tiện) và được cập nhật thường xuyên.
* Hệ thống thương mại điện tử: là các hệ thống website đặc biệt phục vụ việc trao đổi mua bán hàng hóa, các dịch vụ trên môi trường internet. Hệ thống thương mại điện tử bao gồm các nền tảng hỗ trợ các giao thức mua bán, các hình thức thanh toán, chuyển giao hàng….
* Hệ thống điều khiển: là các hệ thống phần mềm gắn với các thiết bị phần cứng hoặc các hệ thống khác nhằm mục đích điều khiển và giám sát hoạt động của thiết bị hay hệ thống đó.

1. 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 | Phát triển phần mềm hướng đối tượng |
| Ưu điểm | * Đặc trưng: phân chia chương trình thành nhiều chương trình con, mỗi chương trình con nhằm đến thực hiện một công việc xác định * Thiết kế dựa trên 2 hướng: hướng dữ liệu hoặc hướng hành động * Tư duy phân tích thiết kế rõ rang, chương trình sáng sủa dễ hiểu | * Tập trung vào cả 2 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 | * Không hỗ trợ việc sử dụng lại * Không phù hợp cho phát triển các phần mềm lớn | * Tốc độ chậm * Khó cho người mới bắt đầu |

1. Trình bày ngắn gọn về quy trình phát triển RUP? Nêu ưu và nhược điểm của mô hình phát triển này?

* Quy trình phát triển hướng đối tượng, được áp dụng rộng rãi.
* RUP được mô tả từ 3 khung nhìn:

• Khung nhìn động -> chỉ ra các pha của mô hình RUP theo chiều thời gian.

• Khung nhìn tĩnh -> chỉ ra các hoạt động trong quy trình RUP.

• Khung nhìn thực tiễn -> đề xuất bộ kinh nghiệm thực hành tốt được sử dụng trong quy trình RUP.

* Ưu nhược điểm của mô hình RUP

Ưu điểm:

* Thường xuyên nhận thông tin phản hồi từ các bên liên quan  
   Những người thực hiện dự án có thể học hỏi tích lũy thêm nhiều kinh nghiệm qua các dự án
* Sử dụng hiệu quả các nguồn lực   
  Sử dụng các kiến trúc thành phần (component), có tính kế thừa ở mức cao, tăng khả năng tái sử dụng
* Bạn có thể cung cấp chính xác những gì khách hàng muốn   
  Các thay đổi có thể quản lý được một cách tốt hơn. Cho phép thay đổi các yêu cầu, các phương thức cho thích hợp hơn
* Các vấn đề được phát hiện sớm  
  Sản phẩm có chất lượng tốt hơn
* Cải thiện khả năng kiểm soát   
  Kiểm tra chất lượng phần mềm liên tục
* Cải thiện quản lý rủi ro  
  Các rủi ro sớm được phát hiện và giảm nhẹ trong khi tích hợp hệ thống và xây dựng dần dự án

Nhược điểm:

* Quá trình có thể quá phức tạp để thực hiện
* Phát triển có thể không kiểm soát được
* Nó là một quá trình nặng
* Cần chuyên gia để hoàn toàn thông qua quá trình này

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 là ngôn ngữ dùng để:

* Mô hình hóa: UML là phương tiện để diễn đạt các mô hình hệ thống dựa trên sự tổ hợp các từ vựng có sẵn của UML theo tập quy tắc xác định
* Biểu diễn:
* Sự giao tiếp giữa các mô hình mức khái niệm dễ phát sinh lỗi trừ khi những người liên quan dùng chung một ngôn ngữ diễn đạt
* Một số yếu tố trong hệ thống phần mềm cần phải mô hình hoá mới có thể hiểu được.
* Một số thông tin sẽ bị mất đi (không được cài đặt) nếu nhóm phát triển không tài liệu hoá các mô hình đang hình dung.
* Đặc tả:
* UML mô tả vấn đề một cách chính xác, không nhập nhằng.
* UML hỗ trợ đặc tả phân tích, thiết kế và các giải pháp cài đặt trong vòng đời phát triển hệ thống.
* Xây dựng tài liệu
* Một số biểu đồ UML có thể được chuyển thành mã nguồn của nhiều ngôn ngữ lập trình khác nhau (Java, C++ …)
* Tăng khả năng tự động hoá trong quá trình phát triển phần mềm
* Tài liệu hóa:
* Các biểu đồ UML đóng vai trò là các tài liệu phát triển hệ thống.
* UML hỗ trợ tạo tài liệu kiến trúc hệ thống cùng các chi tiết của nó.
* UML cho phép diễn đạt các yêu cầu.
* UML cung cấp ngôn ngữ để mô hình hoá các hoạt động trong dự án cũng như việc quản lý chuyển giao phiên bản.

- UML còn được sử dụng để mô hình hoá các hệ thống phi phần mềm như luồng công việc hay cho hoạt động thiết kế phần cứng

- Các phần tử mô hình trong UML:

* Phần tử cấu trúc
* Phần tử hành vi
* Phần tử nhóm gộp
* Phần tử chú thích

Câu 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??

* Có 8 loại biểu đồ:

- Biểu đồ ca sử dụng: Biểu diễn tập các ca sử dụng, các tác nhân và quan hệ giữa chúng.

-> Giải quyết khung nhìn ca sử dụng tĩnh của hthong

- Biểu đồ tương tác: Biểu diễn 1 tương tác gồm tập các đối tượng, quan hệ và các thông điệp giữa chúng.

-> Biều diễn khung nhìn động của hệ thống.

- Biểu đồ lớp: Biểu diễn 1 tập các lớp, các giao diện, các cộng tác và các quan hệ giữa chúng.

-> Biểu diễn khung nhìn thiết kế tĩnh của hthong.

- Biểu đồ đối tượng: Biểu diễn 1 tập các đối tượng và các qhe giữa chúng. Tái hiện hình chụp của hệ thống (snapshorts)

-> Biểu diễn khung nhìn thiết kế tĩnh của hệ thống, phản ảnh các hiện thực trong thế giới thực

- Biểu đồ trạng thái: Biểu diễn 1 máy trạng thái: trang thái, sự kiện, hoạt động, ...

-> Giải quyết khung nhìn động của hthong.

- Biểu đồ hoạt động: Là Trường Hợp đặc biệt của biểu đồ trạng thái, diễn đạt luồng hoạt động trong hệ thống.

-> Mô tả khung nhìn động của hệ thống.

- Biểu đồ thành phần: Biểu diễn cách tổ chức các thành phần và sự phụ thuộc giữa chúng

-> Mô tả khung nhìn cài đặt tĩnh của hệ thống.

- Biểu đồ triển khai: Biểu diễn cấu hình của các nút xử lý và các thành phần thực thi của nút.

->Mô tả khung nhìn triển khai tĩnh của 1 kiến trúc.

\* Cơ sở xác định các loại biểu đồ trên

- Biểu đồ ca sử dụng: đùng để tổ chức và mô hình hóa hành vi của hthong.

- Biểu đồ tương tác: biểu diễn trình tự thời gian hoặc tổ chức cấu trúc của các đối tượng.

- Biểu đồ lớp: dùng trong mhinh hóa các hthong hướng đối tượng.

- Biểu đồ đối tượng: góc nhìn phản ánh các hiện thực trong thế giới thực

- Biểu đồ trạng thái: mô hình hóa hành vi của 1 giao diện .

- Biểu đồ hoạt động: mhinh hóa chức năng của hthong.

- Biểu đồ thành phần: liên quan đến các biểu đồ lớp.

- Biểu đồ triển khai: liên quan bđ thành phần theo cách một nút xử lý.

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 ?

* Khung nhìn tĩnh:

- Mô hình hoá các khái niệm trong miền ứng dụng một cách logic.

- Không mô tả hành vi hệ thống.

- Các thành phần chính của khung nhìn tĩnh là các lớp và mối quan hệ giữa chúng (association, generalization, dependency, realization).

- Khung nhìn tĩnh được thể hiện trong các biểu đồ lớp.

* Khung nhìn thiết kế:

- Khung nhìn thiết kế mô hình hoá cấu trúc thiết kế của ứng dụng. - Khung nhìn thiết kế ánh xạ các lớp vào các thành phần thực thi. - Khung nhìn thiết kế bao gồm:

• Biểu đồ cấu trúc bên trong.

• Biểu đồ cộng tác.

• Biểu đồ thành phần.

* Khung nhìn ca sử dụng:

- Khung nhìn ca sử dụng biểu diễn các chức năng và môi trường dự kiến của hệ thống dưới góc nhìn của người dùng cuối.

- Khung nhìn ca sử dụng nắm bắt các hành vi của hệ thống.

* Khung nhìn máy trạng thái:

- Một máy trạng thái (state machine):

• Mô hình hoá lịch sử vòng đời có thể có của một đối tượng.

• Chứa các trạng thái được kết nối với nhau bởi các chuyển tiếp (transitions).

• Mỗi trạng thái mô hình hoá một khoảng thời gian trong vòng đời của một đối tượng.

• Khi một sự kiện xảy ra -> kích hoạt một chuyển tiếp -> chuyển đối tượng sang một trạng thái mới.

* Khung nhìn hoạt động:

- Biểu đồ hoạt động là trường hợp đặc biệt của biểu đồ trạng thái.

- Biểu diễn luồng hoạt động trong hệ thống.

- Biểu đồ hoạt động chứa các nút hoạt động được kết nối với nhau bởi các luồng điều khiển.

- Biểu đồ hoạt động nhấn mạnh luồng điều khiển giữa các đối tượng.

* Khung nhìn tương tác:

- Một tương tác là hành vi được mô tả bằng cách chỉ ra chuỗi thông điệp trao đổi (truyền và nhận) giữa các đối tượng.

- Các đối tượng tương tác với nhau để thực hiện một kịch bản.

- Khung nhìn tương tác mô tả trình tự trao đổi thông điệp giữa các đối tượng của hệ thống.

- Khung nhìn tương tác cung cấp một cái nhìn toàn diện về hành vi của hệ thống, chỉ ra luồng điều khiển trên nhiều đối tượng.

- Khung nhìn tương tác được thể hiện trong:

• Biểu đồ tuần tự.

• Biểu đồ giao tiếp.

* Khung nhìn triển khai:

- Khung nhìn triển khai biểu diễn cấu hình của các nút xử lý và các chế tác trên các nút xử lý đó tại run-time.

• Một chế tác (artifact) là một thành phần thực thi dạng vật lý, một phần thông tin được sử dụng hoặc được tạo ra trong quy trình phát triển phần mềm.

• Một nút (node) là một tài nguyên tại run-time, biểu diễn một thực thể vật lý của hệ thống (máy tính, bộ nhớ, thiết bị …).

- Biểu đồ triển khai mô tả khung nhìn triển khai tĩnh của một kiến trúc.

* Khung nhìn quản lý mô hình:

- Khung nhìn quản lý mô hình mô hình hoá việc tổ chức của các mô hình được tạo ra.

- Bao gồm một tập các gói, mỗi gói chứa các phần tử mô hình (các lớp, các ca sử dụng …).

- Một gói có thể chứa nhiều gói khác.

- Thông tin quản lý mô hình thường được hiển thị trên biểu đồ gói.

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 vđe trong ptpmhdt:

-Yêu cầu ko đc đáp ứng, thay đổi quá nhanh.

-Các modun ko khớp.

-Phát hiện lỗi muộn, bảo trì kém.

-Chất lượng kém, xung đột người phát triển.

-> Thường bắt nguồn từ 2 căn nguyên: Giao tiếp nhập nhằng và tồn tại phi nhất quán.

\* ND bộ kinh nghiệm thực tiễn:

- Phát triển lặp: + Giải quyết các rủ ro lớn trước khi đầu tư.

+ Sớm nhận đc phản hồi của người dùng.

+ Kiểm thử diễn ra liên tục.

+ Định nghĩa các mốc ngắn hạn cho dự án

-Quản lý yêu cầu: + Phân tích vấn đề.

+ Hiểu sự mong đợi của người dùng.

+ Định nghĩa hệ thống, quản lý phạm vi.

+ Qly thay đổi yêu cầu.

- Sử dụng kiến trúc thành phần: Để điều khiển quy trình phát triển lặp, linh động với sự thay đổi. Có thể bảo trì và mở rộng được.

- Mô hình hóa trực quan: Đơn giản hóa các mô hình phức tạp, dễ dàng quản lý mô hình. Tăng tính trao đổi giữa các thành viên trong nhóm phát triển.

- Kiểm tra chất lượng thường xuyên: Các phép đo và tiêu chuẩn để chứng minh chất lượng đạt được.

cthử, mã nguồn chương trình.

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

* Mục đích:
  + Tìm tác nhân nghiệp vụ: Cá nhân, tổ chức, cơ quan, hệ thống tương tác với nghiệp vụ đó.

VD: Khách hàng, đối tác, nhà cung cấp, nhà đâu tư,...

* + Tìm các ca sử dụng nghiệp vụ: Từ tác nhân nghiệp vụ ta phân tích các giá trị thu được của chúng từ các hoạt động nghiệp vụ. Mỗi tiến trình như 1 ca sử dụng nghiệp vụ.

VD: Ta có các ca sử dụng nghiệp vụ tương ứng với các hoạt động phát triển và cung cấp thông tin nghiệp vụ cho chủ nhân và các nhà đầu tư.

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

Không tìm thấy///

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

Tài liệu đặc tả bổ sung bao gồm các yêu cầu quan trọng cho ca sử dụng, những yêu cầu không được đề cập trong mô hình ca sử dụng. Tài liệu này với tài liệu ca sử dụng 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 phân lớp như sau:

• Ràng buộc chức năng: Gồm các chức năng được dùng chung tham chiếu đến các ca sử dụng khác nhau

• Tính có thể sử dụng được: Các yêu cầu liên quan đến khả năng sử dụng được của hệ thống. Chẳng hạn, các yêu cầu về tính dễ sử dụng hoặc các yêu cầu về đào tạo người dùng trong đó chỉ rõ cách mà người dùng với tư cách là các tác nhân tương tác với hệ thống

• Tính có thể tin cậy được: Bao gồm bất cứ yêu cầu nào liên quan đến tính tin cậy của hệ thống. Nó thường bao gồm các phát biểu về các độ đo, sự lượng hóa như thời gian trung bình giữa các lần thất bại hay số lỗi trên nghìn dòng lệnh

• Tính trình diễn: Bao gồm các ràng buộc về đặc điểm thực thi của hệ thống như các yêu cầu về thời gian phản hồi của hệ thống. Những ràng buộc này thường được phát biểu gắn liền với từng kịch bản ca sử dụng của hệ thống

• Các ràng buộc thiết kế: Bao gồm bất kỳ yêu cầu, ràng buộc về hệ thống đang được xây dựng

Các đợt tạo bổ sung đi kèm với mô hình ca sử dụng. Chúng được tạo ra và xem xét từ rất sớm trong pha xây dựng 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 định nghĩa các thuật ngữ trong miền vấn đề của hệ thống , mô tả ở dạng văn bản

**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 của việc xác định yêu cầu

• giúp hiểu rõ hơn về hệ thống

• xác định được biên của hệ thống

• ước tính đươc chi phí

• định hình được giao diện làm rõ được nhu cầu và mục tiêu của người dùng

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

**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 ?**

-Mục đích của phân tích thiết kế :

+Xác định một kiến trúc tiêu biểu cho hệ thống dựa trên kinh nghiệm thu được từ các hệ thống tương tự hoặc các miền vấn đề tương tự

+Xác định các mẫu kiến trúc, các cơ chế chính và các quy ước về mô hình hệ thống

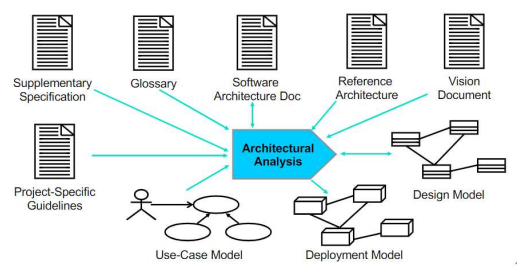
+Xác định chiến lược sử dụng lại

+Cung cấp một phần đầu vào cho hoạt động lập kế hoạch dự án

- Chế tác INPUT/ OUTPUT là đầu vào và ra cho hoạt động phân tích kiến trúc

+Đầu vào: mô hình ca sử dụng, đặc tả bổ sung, từ điển thuật ngữ, mô hình thiết kế, mô hình kiến trúc, các loại tài liệu

+Đầu ra: tlieu ktruc hệ thống, mhinh thiết kế và triển khai.



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

Kiến trúc phần mềm của một chương tình máy tính hay hệ thống là cấu trúc của các thành phần trong hệ thống đó . Kiến trúc phần mềm bao gồm các phẩn tử phần mềm , các thuộc tính và mối quan hệ giữa chúng . Ngoài ra cũng đề cập đến các tài liệu phần mềm hệ thống , thuận tiện cho việc trao đổi thông giữa các thành viên của nhóm dự án

Khung nhìn “4+1” phản ảnh hệ thống từ 4+1 khung nhìn khác nhau:

-Khung nhìn ca sử dụng : định hướng, chi phối các khung nhìn còn lại

-Khung nhìn logic: phân tích kiến trúc

+Khung nhìn tiến trình: mô tả kiến trúc thực thi

+Khung nhìn triển khai: mô tả sự phân tán

+Khung nhìn thực thi: thực thi

Câu 16:

* Vai trò của pha phân tích:

- 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

trong miền quan tâm của bài toán.

- Đưa ra hướng giải quyết bài toán.

* Các bước trong phân tích hướng đối tượng:

Phân tích hướng đối tượng được chia làm ba bước tương ứng với ba dạng mô hình UML là:

- Mô hình use case: bước này nhằm xây dựng mô hình chức năng của sản

phẩm phần mềm.

- Mô hình lớp: biểu diễn các lớp, các thuộc tính và mối quan hệ giữa các lớp.

- Mô hình động: biểu diễn các hoạt động liên quan đến một lớp hay lớp con.

Câu 17:Mục đích hoạt động của phân tích kiến trúc?

Phân tích kiến trúc hướng đến những mục đích sau: thứ nhất, phân tích kiến trúc là để xác định một kiến trúc tiêu biểu cho hệ thống dựa vào kinh nghiệm từ các hệ thống tương tự hay các miền vấn đề tương tự. Thứ hai, phân tích kiến trúc là để xác định các mẫu kiến trúc, các cơ chế kiến trúc chính và và các quy ước về mô hình hệ thống. Thứ ba, phân tích kiến trúc là để xác định các chiến lược sử dụng lại. Và mục đích nữa của phân tích kiến trúc là để cung cấp phần đầu vào cho hoạt động lập kế hoạch dự án.

18.Cơ chế phân tích là gì? Cho ví dụ?

Cơ chế phân tích tái hiện một mẫu về giải pháp chung cho một vấn đề thường gặp. Những cơ chế này có thể là các mẫu về kiến trúc hay vi hay cả hai. Chúng được sử dụng trong pha phân tích để giảm mức độ phức tạp trong phân tích và tăng khả năng duy trì sự nhất quán trong thiết kế. Cơ chế phân tích được sử dụng như một hình thức “khoanh vùng”, “đặt chỗ” ở những tầng trên của kiến trúc cho những hành vi phức tạp trong các tầm thấp hơn. Theo cách đó người phân tích tránh sa vào các chi tiết không cần thiết của những hành vi phức tạp đó. Cơ chế phân tích thường là kết quả từ các thể hiện của các mẫu phân tích hay mẫu thiết kế.

Cơ chế phân tích thường ít liên quan đến vấn đề mà chủ yếu là các khái niệm trong “khoa học máy tính”. Cho nên chúng thường hướng đến các hành vi ở các tầng giữa và tầng dưới của kiến trúc. Chúng cung cấp các hành vi chuyên biệt cho các lớp miền hay các thành phần hoặc đưa ra các bản cài đặt về cộng tác giữa các lớp hay các thành phần.

Một số ví dụ:

Cơ chế bền vững: giải quyết vấn đề về tổ chức bền vững

Cơ chế giao tiếp tiến trình: giải quyết vấn đề về tổ chức giữa các tiến trình

Cơ chế truyền thông điệp: giải quyết các vấn đề về tổ chức định hướng và truyền thông điệp

Cơ chế phân tán: giải quyết các vấn đề về tổ chức thành phần trên nhiều nút xử lý của hệ thống

Cơ chế quản lý giao dịch: giải quyết các vấn đề về tổ chức quản lý giao dịch

Cơ chế điều khiển và đồng bộ hóa tiến trình: giải quyết vấn đề về cơ chế đồng bộ giữa các tiến trình

Cơ chế an ninh: giải quyết các vấn đề về an ninh hệ thống

Mỗi cơ chế thường được xem xét trên những đặc điểm chính nào đó. Chẳng hạn, với cơ chế lưu trữ bền vững, chúng ta cần xem xét khía cạnh, đặc điểm sau:

Phạm vi: xác định tất cả các lớp mà thể hiện của chúng cần được lưu trữ cho các lần thực thi sau của hệ thống

Quy mô: xác định miền số lượng các đối tượng cần lưu trữ bền vững

Dung lượng: số lượng tối ta các đối tượng cần lưu trữ bền vững

Thời lượng: khoảng thời gian mà các đối tượng cần lưu trữ bền vững

Tần suất truy cập: cần phải xác định xem đối tượng có từ xuyên cập nhật, thay đổi không không, hay là hầu như không bị thay đổi

Độ tin cậy: vấn đề đặt ra là các đối tượng có cần phải bảo toàn những lỗi và xung đột giữa các tiến trình không

19.Mục đích hoạt động phân tích ca sử dụng là gì?

Mục đích của tài liệu đào tạo bổ sung ca sử dụng là để nắm bắt thêm các thông tin để hiểu được hành vi bên trong của hệ thống, những hành vi mà trong phần mô tả ca sử dụng viết cho khách hàng có thể không được đề cập. Thông tin này được sử dụng như đầu vào cho các bước còn lại của hoạt động phân tích ca sử dụng. Thông tin đó cũng là cơ sở cho việc gắn trách nhiệm cho các lớp phân tích. Lưu ý trong một vài trường hợp, chúng ta có thể thấy rằng, một vài yêu cầu là không chính xác hoặc không thể hiểu được. Trong những trường hợp đó luồng kịch bản các sự kiện của các ca sử dụng cần phải được cập nhật, chẳng hạn, bằng cách quay lặp lại khâu nắm bắt yêu cầu.

Câu 20:

* Lớp phân tích:
  + Các lớp phân tích tái hiện một mô hình khái niệm cho những phần tử có trách nhiệm và hành vi trong hệ thống.
  + Các lớp phân tích vận dụng các yêu cầu chức năng chính.
  + Các lớp phân tích mô hình hóa các đối tượng trong miền vấn đề
* Phân loại các lớp phân tích dựa trên 3 khía cạnh của hệ thống:
  + Biên giữa hệ thống và các tác nhân-> lớp biên(stereotype dạng<<boundary>>)
  + Thông tin hệ thống sử dụng -> lớp thực thể (stereotype dạng <<entity>>)
  + Logic điều khiên của hệ thống -> lớp điều khiển (stereotype dạng <<control>>)
* Vai trò và cách xác định:
  + Lớp biên:
    - Vai trò:

- Cầu nối giao tiếp giữa giao diện và những thứ bên ngoài hệ thống (môi trường)

- Mô hình hóa tương tác giữa môi trường hệ thống và phần hoạt động bên trong HT

* + - Cách xác định:

- Lớp UI, lớp giao diện hệ thống (form,report…)

- khảo sát biểu đồ UC để tìm kiếm lớp biên

- mỗi cặp tác nhân và UC sẽ tương ứng với một lớp biên (khuyến cáo)

* + Lớp thực thể:
    - Vai trò:

- Là lớp lưu trữ thoong tin sẽ ghi vào bộ nhớ ngoài

- thông thường phải tạo ra một bảng CSDL cho lớp thực thể

* + - Cách xác định:

-Có thể được xác định dựa vào mô tả luồng sự kiện của UC

* + Lớp điều khiển:
* Vai trò:

- Điều phối các hoạt động của các lớp khác

- Mỗi UC chỉ nên có một lớp điều khiển ở khâu phân tích

- Làm mịn thành nhiều lớp ở khâu thiết kế.

* + - Cách xác định:

- Không thực hiện chức năng nghiệp vụ nào

- Có thể điều khiển sự kiện liên quan đến an ninh hệ thoongs hoặc giao dịch CSDL.

Câu 21:

|  |  |
| --- | --- |
| Biểu đồ tuần tự | Biểu đồ cộng tác |
| -Biểu diễn mối quan hệ giữa các đối tượng và giữa các đối tượng với các tác nhân theo thứ tự thời gian.  -Có các thành phần cơ bản:  +Các đối tượng (object):không cùng xuất hiện ở trên cùng của biểu đồ mà chúng chỉ xuất hiện (về mặt thời gian) khi thực sự tham gia vào tương tác.  +Các message: Được biểu diễn dưới dang phi hình thức hoặc dưới dạng hình thức. Được cho phép từ một đối tượng tới chính bản thân nó  + Đường lifeline: là một đường kẻ nối dài phía dưới đối tượng, mô tả quá trình của một đối tượng trong tương tác thuộc biểu đồ.  +Chú thích: giúp người đọc dễ dàng hiểu được nội dung biểu đồ đó. | -Biểu diễn mqh giữa các đối tượng và các đối tượng với các tác nhân nhưng nhấn mạnh đến vai trò của các đối tượng trong tương tác.  -Có các thành phần cơ bản:  +Các đối tượng: được đặt tự do trong không gian của biểu đồ. Các đối tượng tham gia tương tác luôn xuất hiện tại một vị trí xác định  +Các message: được đánh số theo thứ tự xuất hiện trong kịch bản mô tả use case tương ứng  +Đường lifeline: không có  +Các liên kết: giữa hai đối tượng có tương tác sẽ có một liên kết nối hai đối tượng đó. Liên kết này không có chiều |

Câu 22:

|  |  |
| --- | --- |
| Include | Extend |
| -Một ca sử dụng có thể tích hợp hành vi của các ca sử dụng khác như là các phần trong hành vi tổng thể của nó  -Biểu diễn một use-case chứa  hành vi được định nghĩa trong  một use-case khác  -Cho phép một use-case sử dụng chức năng của use-case khác  -Sử dụng stereotype là  <<include>> | -Một ca sử dụng có thể được  định nghĩa như là một sự mở  rộng tăng dần của một ca sử  dụng cơ sở  -Chèn hành vi của ca sử dụng  mở rộng vào ca sử dụng cơ sở  -Cho phép mở rộng chức năng  của một ca sử dụng  -Sử dụng stereotype là  <<extend>> |

Câu 23:

* Quan hệ liên kết (Association):

 - Một quan hệ ngữ nghĩa giữa hai hoặc nhiều lớp có mối liên hệ với

nhau giữa các thể hiện của chúng.

- Một quan hệ cấu trúc, chỉ ra các đối tượng của lớp này có kết nối

với các đối tượng của lớp khác hoặc chính lớp đó.

 - Một liên kết giữa hai lớp chỉ ra rằng đối tượng ở một đầu của liên

kết nhận ra đối tượng ở đầu kia và có thể gửi thông điệp cho nhau.

* Quan hệ kết tập (aggregation):

- Một dạng đặc biệt của liên kết.

- Mô hình hoá mối quan hệ toàn thể-bộ phận (whole-part) giữa đối

tượng toàn thể và các bộ phận của nó.

- Mối quan hệ “là một phần của” (“is a part of”)

* Quan hệ hợp thành (composition):

- Một dạng đặc trưng của kết tập với quyền sở hữu mạnh và các

vòng đời trùng khớp giữa hai lớp.

* Whole là sở hữu duy nhất của Part, tạo và huỷ Part.
* Part bị bỏ khi Whole bị bỏ, Part không thể tồn tại nếu Whole

không tồn tại.

* Quan hệ tổng quát hóa (generalization): Quá trình bắt đầu từ một lớp

chuyên biệt hóa đến lớp có mức khái quát cao hơn được gọi là quá

trình tổng quát hóa.

* Quan hệ phụ thuộc: Một quan hệ ngữ nghĩa giữa hai lớp, trong đó sự

thay đổi của lớp này kéo theo sự thay đổi của lớp kia (lớp phụ thuộc)

mặc dù giữa chúng không có một sự liên kết rõ ràng

Câu 24:

* Vai trò của pha thiết kế:
  + 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 bước trong 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:

Câu 27:

Cơ chế phân tích trong thiết kế hướng đối tượng dùng để làm gì?

Trong phân tích kiến trúc, việc xác định các cơ chế phân tích cho hệ thống đang phát triển là rất quan trọng.

- Cơ chế phân tích tập trung và giải quyết các yêu cầu phi chức năng của hệ thống chẳng hạn như các yêu cầu về lưu trữ bền vững, độ tin cậy và khả năng trình diễn.

- Giải pháp cho các yêu cầu phi chức năng qua các cơ chế phân tích sẽ được tích hợp trực tiếp vào kiến trúc.

- Phương thức tổ chức các cơ chế phân tích như trên sẽ góp phần làm giảm đáng kể độ phức tạp của kiến trúc.

- Nó cũng góp phần đảm bảo tính nhất quán trong thiết kế vì đã cung cấp một cơ chế tham chiếu đến nhóm hành vi đối tượng phức tạp, được tổ chức một cách độc lập.

- Cơ chế phân tích cũng cho phép người phân tích tập trung tốt hơn cho việc chuyển các yêu cầu chức năng thành các trừu tượng phần mềm, bởi vì họ không bị phân tâm và sa vào việc giải quyết các yêu cầu phi chức năng vốn là một thứ yếu trong giai đoạn này.

Tại sao cần phân loại các cơ chế phân tích?

Câu 26: Hệ thống con là gì?

Một hệ thống con là một phần tử mô hình có ngữ nghĩa giống với một gói, ví dụ nó có thể chứa các phần tử mô hình, lớp. Một hệ thống con thực thi một hoặc nhiều giao diện dùng để định nghĩa các hành vi của hệ thống con có thể thực hiện. Hệ thống con có thể biểu diễn bằng ký hiệu của ngôn ngữ UML của một gói với kiểu mở rộng << subsystem>>

Phân biệt hệ thống con và gói?

Hệ thống con và gói khá giống nhau, tuy nhiên chúng cũng khác biệt ở một số đặc điểm.

|  |  |
| --- | --- |
| Hệ thống con | Gói |
| - cung cấp các giao diện chứa các hành vi mà nó có thể truy cập được | - gói không cung cấp các hành vi mà nó là các bộ chứa trong đó có các hành vi |
| - đóng gói hoàn toàn nội dung của nó  - chỉ cung cấp các hành vi thông qua các giao diện  - Quan hệ phụ thuộc trong hệ thống con là quan hệ trên giao diện | - tổ chức và điều khiển tập các lớp cần thiết thông thường  - chỉ được sử dụng để mô hình hóa sự tổ chức và quản lý cấu hình  - Quan hệ phụ thuộc trong gói là trên các phần tử của gói |

Trình bày mục đích sử dụng và cách xác định hệ thống con?

* Chúng ta sử dụng hệ thống con theo đơn vị chuyển giao, theo cấu hình, theo thứ bậc khi hệ thống hoàn thành.
* Việc cung cấp các tài nguyên và khả năng của các nhóm lập trình đòi hỏi dự án được phân chia theo các nhóm khác nhau
* Nhiều yêu cầu bị thay đổi bắt nguồn từ người dùng -> các hệ thống con đảm bảo các thay đổi này chỉ ảnh hưởng đến phần của hệ thống tương ứng với phần của người sử dụng đó

Câu 28: Mục đích của thiết kế ca sử dụng là gì?

* Là để làm mịn hiện thực hóa ca sử dụng và các yêu cầu liên quan đến thao tác của các lớp và hệ thống con
* Các chế tác đầu vào của hoạt động này bao gồm mô hình phân tích, các đặc tả bổ sung, các lớp thiết kế, hệ thống con,… và kết quả của hoạt động này là một mô hình thiết kế với các ca sử dụng đã hiện thực hóa ở mức thiết kế.

Câu 29: Mục đích của việc thiết kế lớp là gì?

* Để đảm bảo các lớp cung cấp các hành vi mà Hiện thực hóa ca sử dụng yêu cầu, đảm bảo cung cấp các thông tin đầy đủ để cài đặt lớp
* 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.

Các lớp được làm mịn trong Thiết kế hướng đối tượng bằng cách nào?

Các lớp thiết kế cần được làm mịn để nắm bắt các yêu cầu phi chức năng đã mô tả trong tài liệu hướng dẫn thiết kế của dự án. Đầu vào quan trọng của bước này là các yêu cầu phi chức năng trong lớp được mô tả trong các trách nhiệm và yêu cầu đặt biệt. Các yêu cầu này thường mô tả theo cách cơ chế kiến trúc nào cần để thực hiện lớp.

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

Process vs Thread

* Để tạo nhiều thread thì dễ dàng hơn so với process vì chúng không cần các địa chỉ nhớ riêng rẽ.
* Các thread chia sẻ các cấu trúc chung mà chỉ sử dụng được bởi từng thread vào mỗi thời điểm. Khác với thread, các process không dùng chung địa chỉ nhớ.
* Thread sử dụng ít tài nguyên hơn so với các process.
* Các process chạy độc lập với nhau. Các thread thì sử dụng chung các địa chỉ nhớ liên kết với nhau.
* Một process có thể chứa nhiều thread.

Operation vs Method