1. Phân tích thiết kế hướng đối tượng là gì? trình bày các thành phần cơ bản của đối tượng?

a)Khái niệm OOAD:(Có thể nêu cách khác ở cuối trang)

Phân tích và thiết kế hướng đối tượng là một kỹ thuật tiếp cận phổ biến dùng để phân tích, thiết kế một ứng dụng, hệ thống. Nó dựa trên bộ các nguyên tắc chung, đó là một tập các hướng dẫn để giúp chúng ta tránh khỏi một thiết kế xấu. 5 nguyên tắc SOLID trong thiết kế hướng đối tượng:

1. Một lớp chỉ nên có một lý do để thay đổi, tức là một lớp chỉ nên xử lý một chức năng đơn lẻ, duy nhất thôi. Nếu đặt nhiều chức năng vào trong một lớp sẽ dẫn đến sự phụ thuộc giữa các chức năng với nhau và mặc dù sau đó ta chỉ thay đổi ở một chức năng thì cũng phá vỡ các chức năng còn lại.
2. Các lớp, module, chức năng nên dễ dàng Mở (Open) cho việc mở rộng (thêm chức năng mới) và Đóng (Close) cho việc thay đổi.
3. Lớp dẫn xuất phải có khả năng thay thế được lớp cha của nó.
4. Chương trình không nên buộc phải cài đặt một interface mà nó không sử dụng đến.
5. Các module cấp cao không nên phụ thuộc vào các module cấp thấp. Cả hai nên phụ thuộc thông qua lớp trừu tượng. Lớp trừu tượng không nên phụ thuộc vào chi tiết. Chi tiết nên phụ thuộc vào trừu tượng

b)Các thành phần cơ bản của đối tượng:

**Trạng thái (state)**: Những đặc điểm của một đối tượng. Trạng thái của một đối tượng thường sẽ thay đổi theo thời gian, và nó được định nghĩa qua một tổ hợp các thuộc tính, với giá trị của các thuộc tính này cũng như mối quan hệ mà đối tượng có thể có với các đối tượng khác. Chẳng hạn như đối tượng Sinh viên thì có mã, họ tên, ngày tháng năm sinh,...

**Hành vi (behavior)**: Những hành động mà một đối tượng thực hiện.Hành vi được thực thi qua loạt các Phương thức (operation) của đối tượng.Ví dụ: Sinh viên thì có các hành vi như đi học, đi dã ngoại vui chơi giải trí,...

**Sự nhận diện (Identity):** đảm bảo rằng mỗi đối tượng là duy nhất – dù trạng thái của nó có thể giống với trạng thái của các đối tượng khác. Ví dụ, khóa học đại số 101 chương 1 và khóa học đại số 101 chương 2 là hai đối tượng trong hệ thống ghi danh trường học. Mặc dù cả hai đều thuộc loại bảng ghi danh, mỗi khóa học vẫn có sự nhận dạng duy nhất của mình.

1. Trình bày các giai đoạn phát triển hệ thống hướng đối tượng?

**+Phân tích hướng đối tượng (Object Oriented Analysis -OOA)**

▪ Giai đọan phát triển một mô hình chính xác và súc tích của vấn đề  
 ▪ Thành phần là các đối tượng và khái niệm đời thực  
 ▪ Vấn đề được trình bày bằng các thuật ngữ tương ứng với các  
 đối tượng có thực  
 ▪ Mô hình thiết kế sẽ chứa các thực thể trong một vấn đề có thực  
 và giữ nguyên các mẫu hình về cấu trúc, quan hệ cũng như  
 hành vi của chúng

▪ Ví dụ: một phòng bán ô tô sẽ có các thực thể:  
 • Khách hàng  
 • Người bán hàng  
 • Phiếu đặt hàng  
 • Phiếu (hoá đơn) thanh toán  
 • Xe ô tô

**+Thiết kế hướng đối tượng (Object Oriented Design - OOD):**

▪ Trong giai đoạn OOD, nhà thiết kế định nghĩa  
 • các chức năng, thủ tục (operations),  
 • thuộc tính (attributes)  
 • mối quan hệ của một hay nhiều lớp (class) và quyết định chúng cần  
 phải được điều chỉnh sao cho phù hợp với môi trường phát triển  
 • thiết kế ngân hàng dữ liệu và áp dụng các kỹ thuật tiêu chuẩn hóa HĐT  
 ▪ Cuối giai đoạn OOD đưa ra một loạt các biểu đồ (diagram):  
 • biểu đồ tĩnh biểu thị các lớp và đối tượng  
 • biểu đồ động biểu thị tương tác giữa các lớp và phương thức hoạt  
 động chính xác của chúng.

**+Lập trình hướng đối tượng (Object Oriented Programming - OOP):**

Giai đoạn xây dựng phần mềm có thể được thực hiện sử dụng kỹ thuật lập trình hướng đối tượng. Đó là phương thức thực hiện thiết kế hướng đối tượng qua việc sử dụng một ngôn ngữ lập trình có hỗ trợ các tính năng hướng đối tượng. Một vài ngôn ngữ hướng đối tượng thường được nhắc tới là C++ và Java. Kết quả chung cuộc của giai đoạn này là một loạt các code chạy được, nó chỉ được đưa vào sử dụng sau khi đã trải qua nhiều vòng quay của nhiều bước thử nghiệm khác nhau.

1. Cho biết trong pttk hdt có những biểu đồ nào? Trình bày về khung nhìn trong pttk hdt (thành phần quan trọng).

a)PTTKHDT có các biểu đồ sau:

Các biểu đồ trong UML:

**–          Biểu đồ UseCase:**

* Định nghĩa: biểu đồ Use case chỉ ra một số lượng các tác nhân ngoại cảnh và mối liên kết của chúng đối với Use case mà hệ thống cung cấp
* Tính chất:
  + Một Use case là một lời *miêu tả của một chức năng* mà hệ thống cung cấp
  + Use case được *miêu tả* duy *nhất theo hướng nhìn từ ngoài vào* của các tác nhân
  + *không miêu tả* chức năng được cung cấp sẽ *hoạt động nội bộ* bên trong hệ thốn

**–          Biểu đồ lớp:**

* Định nghĩa: biểu đồ lớp chỉ ra cấu trúc tĩnh của các lớp trong hệ thống
* Tính chất:
  + *Các lớp có thể quan hệ với nhau* trong nhiều dạng thức: liên kết (associated – được nối kết với nhau), phụ thuộc (dependent – một lớp này phụ thuộc vào lớp khác), chuyên biệt hóa (specialized – một lớp này là một kết quả chuyên biệt hóa của lớp khác), hay đóng gói ( packaged – hợp với nhau thành một đơn vị).
  + *đi kèm* với cấu trúc bên trong của các lớp theo khái niệm *thuộc tính (attribute) và thủ tục (operation).*

**–          Biểu đồ đối tượng:**

* Định nghĩa:
  + là một phiên bản của biểu đồ lớp và thường cũng sử dụng các ký hiệu như biểu đồ lớp
  + *khác biệt* giữa hai loại biểu đồ này *nằm ở chỗ*biểu đồ đối tượng *chỉ ra một loạt các đối tượng thực thể* của lớp, thay vì các lớp
* Tính chất:
  + có hai ngoại lệ: *đối tượng* được viết với tên *được gạch dưới* và *tất cả các thực thể*trong một mối quan hệ đều *được chỉ ra*

**–          Biểu đồ trạng thái:**

* Định nghĩa:
  + là một sự bổ sung cho lời miêu tả một lớp
  + Nó *chỉ ra tất cả các trạng thái* mà *đối tượng*của lớp này *có thể có*, và những *sự kiện* (event) nào sẽ *gây ra sự thay đổi trạng thái* (hình 3.5).
* Tính chất:
  + không được vẽ cho tất cả các lớp
  + chỉ riêng cho những lớp có một số lượng các trạng thái được định nghĩa rõ ràng và hành vi của lớp bị ảnh hưởng và thay đổi qua các trạng thái khác nhau

**–          Biểu đồ trình tự:**

* Định nghĩa: Một biểu đồ trình tự *chỉ ra một cộng tác* *động* giữa *một loạt các đối tượng*
* Tính chất:
  + chứa một loạt các đối tượng được biểu diễn bằng các đường thẳng đứng
  + Trục thời gian có hướng từ trên xuống dưới trong biểu đồ, và biểu đồ chỉ ra sự trao đổi thông điệp giữa các đối tượng khi thời gian trôi qua
  + Các thông điệp được biểu diễn bằng các đường gạch ngang gắn liền với mũi tên

**–          Biểu đồ cộng tác:**

* Định nghĩa: biểu đồ cộng tác chỉ ra một sự cộng tác động
* Tính chất:
  + Thường người ta sẽ chọn *hoặc dùng biểu đồ trình tự hoặc dùng biểu đồ cộng tác*
  + sử dụng biểu đồ trình tự hay biểu đồ cộng tác thường sẽ được quyết định theo nguyên tắc chung sau:
    - Nếu *thời gian hay trình tự* là yếu tố *quan trọng* nhất cần phải nhấn mạnh thì hãy chọn *biểu đồ trình tự*
    - nếu *ngữ cảnh* là yếu tố quan trọng hơn, hãy chọn *biểu đồ cộng tác*

**–          Biểu đồ hoạt động**

* Định nghĩa:
  + biểu đồ hoạt động chỉ ra một trình tự lần lượt của các hoạt động
  + miêu tả các *hoạt động* được thực hiện *trong một thủ tục*
* Tính chất:
  + bao gồm các trạng thái hành động

**–          Biểu đồ thành phần:**

* Định nghĩa: biểu đồ thành phần chỉ ra *cấu trúc vật lý của các dòng lệnh*
* Tính chất:
  + Một thành phần chứa các thông tin về các lớp logic hoặc các lớp mà nó thi hành, như thế có nghĩa là nó tạo ra một ánh xạ từ hướng nhìn logic vào hướng nhìn thành phần.

**–          Biển đồ triển khai**

* Định nghĩa: Biểu đồ triển khai chỉ ra ki*ến trúc vật lý của phần cứng*cũng *như phần mềm trong hệ thống*
* Tính chất: chỉ ra hướng nhìn triển khai, miêu tả kiến trúc vật lý thật sự của hệ thống

b)Trình bày về khung nhìn trong phân tích thiết kế hướng đối tượng:

+Hướng nhìn Use Case (Use Case View): đây là hướng nhìn chỉ ra khía cạnh chức năng của một hệ thống, nhìn từ hướng tác nhân bên ngoài.

+Hướng nhìn logic (Logical View): chỉ ra chức năng sẽ được thiết kế bên trong hệ thống như thế nào, qua các khái niệm về cấu trúc tĩnh cũng như ứng xử động của hệ thống.

+Hướng nhìn thành phần (Component View): chỉ ra khía cạnh tổ chức của các thành phần code.

+Hướng nhìn song song (Concurrency View): chỉ ra sự tồn tại song song/ trùng hợp trong hệ thống,hướng đến vấn đề giao tiếp và đồng bộ hóa trong hệ thống.

+Hướng nhìn triển khai (Deployment View): chỉ ra khía cạnh triển khai hệ thống vào các kiến trúc vật lý (các máy tính hay trang thiết bị được coi là trạm công tác).

Có 5 khung nhìn :(Nêu cách khác)

* Use Case View : bao gồm các UC mô tả ứng xử của hệ thống theo cách nhìn nhận của người dùng, người phân tích hệ thống. Nó không chỉ ra cách cấu trúc của hệ thống phần mềm mà chỉ dùng để nhín nhận 1 cách tổng quát những j mà hệ thống sẽ cung 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 lý đồng thời. Dùng cho người phát triển và tích hợp hệ thống bao gồm các biểu đồ sequence, collaboration, activity, và state.
* Implementation View : bao gồm các component và file tạo nên hệ thống vật lý. Nó chỉ ra sự phụ thuộc giữa các thành phần này, cách kết hpwj chúng lại 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.

**Câu 4: Hãy cho biết biểu đồ trình tự là gì, mục tiêu và chức năng của việc xây dựng biểu đồ trình tự? mô tả các thành phần trong biểu đồ trình tự.**

**a)Khái niệm:** Sequence Diagarm là bản vẽ mô tả sự tương tác của các đối tượng để tạo nên các chức năng của hệ thống.

**b)Mục tiêu và chức năng:**

**+Mô tả tương tác giữa các actor và các đối tượng hệ thống.**

**+Mô tả sự tương tác giữa các đối tượng theo trình tự thời gian.**

**+Mô hình hóa khía cạnh động của hệ thống**

**+Thường được dùng để biểu diễn các bước thực hiện trong một kịch bản khai thác**

**(Scenario) của một use case.**

**+Dùng để mô hình hóa giao tác giữa các hệ thống con.**

**+Lập mô hình tương tác ở mức cao giữa các đối tượng hoạt động trong một hệ thống**

**+Được sử dụng để xác định và chỉ rõ vai trò của  
 các đối tượng tham gia vào luồng sự kiện của use  
 case**

**c) Các thành phần của biểu đồ trình tự:**

**-**Axes:

+ Trục tọa độ, trục ngang thể hiện các đối tượng, trục đứng thể hiện thời gian

-Actor:

+Tác nhân bên ngoài tương tác với hệ thống.

-Đối tượng:

+Tham gia quá trình tương tác giữa người dùng và hệ thống.

+Biểu diễn bằng hình chữ nhật bên trong là tên đối tượng.

-Đường sinh tồn:

+Biểu diễn bằng đường gạch thẳng đứt nét cho biết thời gian sống của đối tượng

-Vùng hoạt động:

+Biểu diễn bằng hình chữ nhật hẹp dọc theo đường sinh tồn.

+Cho biết thời gian thực thi một hoạt động tương ứng

-Thông điệp:

+Thể hiện thông điệp từ 1 đối tượng này tương tác với một đối tượng khác

+Là các đường ngang nối liền các đường sinh tồn giữa các đối tượng

**Câu 5: Hãy cho biết biểu đồ trạng thái là gì, mục tiêu và chức năng của việc xây dựng biểu đồ trạng thái? mô tả các thành phần trong biểu đồ trạng thái.**

**a)Khái niệm:**

**Biểu đồ trạng thái mô tả các thông tin về các trạng thái khác nhau của đối tượng, biểu diễn các trạng thái và sự chuyển tiếp giữa các trạng thái của các đối tượng trong 1 lớp xác định, hoạt động của đối tượng trong mỗi trạng thái ra sao.Thông thường, mỗi lớp sẽ có 1 biểu đồ trạng thái.**

**b)Mục tiêu và chức năng:**

**+Mô tả chu trình sống của các đối tượng chính từ khi sinh ra, hoạt động và mất đi.**

**+Chỉ ra hành vi của đối tượng**

**+Biểu diễn mối quan hệ giữa các trạng thái của đối tượng**

**+Sử dụng để mô hình hóa khía cạnh hoạt động của lớp**

**c) Mô tả các thành phần trong biểu đồ trạng thái:**

**+Trạng thái-State**

**+Sự kiện-Event**

**+Hành động-Action**

**+Mối liên hệ giữa các trạng thái**

**Câu 6: Hãy cho biết biểu đồ hoạt động là gì, mục tiêu và chức năng của việc xây dựng biểu đồ hoạt động? mô tả các thành phần trong biểu đồ hoạt động.**

**a)Khái niệm:**

**Biểu đồ hoạt động được sử dụng để mô tả các hoạt động và các hành động được thực hiện trong một use case.**

**b)Mục tiêu và chức năng:**

**+Mô tả các luồng công việc, qui trình nghiệp vụ**

**+Hỗ trợ việc mô tả các xử lý xong xong**

**+Chỉ ra các bước trong luồng công việc**

**+Chỉ ra Các điểm quyết định**

**+Chỉ ra Ai có trách nhiệm thực hiện từng bước**

**+ Chỉ ra Các đối tượng ảnh hưởng đến luồng công việc**

**+Phân tích Use Case**

**+Phân tích nghiệp vụ để hiểu rõ hệ thống**

**+Cung cấp thông tin để thiết kế bản vẽ Sequence Diagram**

**c)Mô tả các thành phần trong biểu đồ hoạt động:**

* **Activity Partition (Swimlines): xác định đối tượng nào tham gia hoạt động nào trong một qui trình.**
* **Fork**

**+Fork thể hiện cho trường hợp thực hiện xong một hoạt động rồi sẽ rẽ nhánh tthực hiện nhiều hoạt động tiếp theo.**

* **Join**

**+Cùng ký hiệu với Fork nhưng thể hiện trường hợp phải thực hiện hai hay nhiều hành động trước rồi mới thực hiện hành động tiếp theo.**

* **Luồng hoạt động**

**+Minh họa quá trình chuyển đổi từ một trạng thái hoạt động này sang trạng thái hoạt động khác.**

* **Branch**

**+Branch thể hiện rẽ nhánh trong mệnh đề điều kiện**

* **Activity**

**+Activity mô tả một hoạt động trong hệ thống. Các hoạt động này do các đối tượng thực hiện.**

* **Nút Start, End**

**+Start thể hiện điểm bắt đầu qui trình, End thể hiện điểm kết thúc qui trình.**

* **Nút Merge**

**+Sự kết hợp của các luồng sự kiện. Các đầu vào không đồng bộ.**

**Nhiều đầu vào và chỉ có một đầu ra**

**Câu 7: Hãy cho biết mục tiêu và chức năng của việc xây dựng biểu đồ use case? Để xây dựng biểu đồ này ta cần phải làm những công việc gì.**

**a)Mục tiêu và chức năng:**

**+Mô tả các chức năng của hệ thống dựa trên quan điểm người dùng.**

**+Mô tả sự tương tác giữa người dùng và hệ thống.**

**+Cho biết hệ thống được sử dụng như thế nào.**

**+Là công cụ mạnh mẽ cho việc lập kế hoạch 🡺 Được dùng cho tất cả các giai đoạn trong quy trình phát triển hệ thống**

**b) Các bước xây dựng Use Case Diagram:**

**Bước 1: Tìm các Actor.**

**+Trả lời các câu hỏi sau để xác định Actor cho hệ thống:**

**Ai sử dụng hệ thống này?**

**Hệ thống nào tương tác với hệ thống này?**

**Bước 2: Tìm các Use Case.**

**Trả lời câu hỏi các Actor sử dụng chức năng gì trong hệ thống? chúng ta sẽ xác  
 định được các Use Case cần thiết cho hệ thống.**

**Bước 3: Xác định các quan hệ.**

**Phân tích và các định các quan loại hệ giữa các Actor và Use Case, giữa các Actor với nhau, giữa các Use Case với nhau sau đó nối chúng lại chúng ta sẽ được bản vẽ Use Case.**

**Câu 8: Hãy cho biết mục tiêu và chức năng của việc xây dựng biểu đồ lớp? Để xây dựng biểu đồ này ta cần phải làm những công việc gì.**

**a)Mục tiêu và chức năng:**

**+Mô tả các đối tượng và mối quan hệ của chúng trong hệ thống.**

**+Mô tả các thuộc tính và các hành vi của đối tượng.**

**+Sử dụng để phân tích thiết kế các chức năng (Sequence Diagram, State Diagram,..)**

**+Sử dụng để cài đặt.**

**b)Các bước xây dựng biểu đồ lớp:**

**Bước 1: Tìm các class dự kiến**

Entity Classes(các lớp thực thể) là các thực thể có thật và hoạt động trong hệ thống, bạn dựa vào các nguồn sau để xác định chúng:

✓**Requirement statement**: Các yêu cầu. Chúng ta phân tích các danh từ trong các  
yêu cầu để tìm ra các thực thể.  
✓**Use Cases**: Phân tích các Use Case sẽ cung cấp thêm các Classes dự kiến.  
✓**Previous và Similar System:** có thể sẽ cung cấp thêm cho bạn các lớp dự kiến.  
✓**Application Experts:** các chuyên gia ứng dụng cũng có thể giúp bạn.

***Lưu ý***: Chỉ các thực thể bên trong hệ thống được xem xét, các thực thế bên ngoài hệ  
thống không được xem xét. Ví dụ Customers là những người khách hàng được quản lý  
trong hệ thống chứ không phải người dùng máy ATM bên ngoài. Bạn phải lưu ý điều này  
để phân biệt Class và Actor.

***Bước 2: Tìm các thuộc tính và phương thức cho lớp***✓**Tìm thuộc tính**: phân tích thông tin từ các form mẫu có sẵn, bạn sẽ tìm ra thuộc  
tính cho các đối tượng của lớp. Ví dụ các thuộc tính của lớp Customer sẽ thể hiện  
trên Form đăng ký thông tin khách hàng.  
✓**Tìm phương thức**: phương thức là các hoạt động mà các đối tượng của lớp này có  
thể thực hiện. Chúng ta sẽ bổ sung phương thức đầy đủ cho các lớp khi phân tích  
Sequence Diagram .

***Bước 3: Xây dựng các quan hệ giữa các lớp và phát hiện các lớp phát sinh***Phân tích các quan hệ giữa các lớp và định nghĩa các lớp phát sinh do các quan hệ  
sinh ra.

1. Năm thuộc tính của hệ thống phức tạp.
   1. Cấu trúc phân cấp.

* Tất cả các hệ thống đều có các hệ thống con và tất cả các hệ thống đều là một phần của các hệ thống lớn hơn.
* Thực tế là nhiều hệ thống phức tạp có khả năng gần như phân tách, cấu trúc phân cấp là yếu tốt tạo điều kiện chính giúp chúng ta hiểu, mô tả, và thậm chí là ‘thấy’ các hệ thống như vậy và các bộ phận của chúng.
  1. Cơ sở (primitive) là tương đối.
* Sự lựa chọn những thành phần nào trong một hệt thống là cơ sở thì mang tính tương đối tùy ý và phần lớn phụ thuộc vào ý kiến của người quan sát hệ thống.
  1. Tách biệt quan hệ.
* Hệ thống phân cấp có thể phân tích được.
* Liên kết nội bộ (intr-component) nói chung là mạnh hơn liên kết inter-component (liên thành phần).
* Tần số cao của các thành phần – liên quan cấu trúc bên trong của các thành phần.
* Từ tần số thấp động – liên quan đến sự tương tác giữa các thành phần.
* Sự khác biệt này giữa các tương tác nội tại và và giữa các phần tử cung cấp một sự rõ ràng tách mối quan tâm giữa các thành phần khác nhau của một hệ thống, làm cho nó có thể nghiên cứu từng phần trong sự cô lập tương đối.
  1. Mẫu chung.
* Hệ thống phân cấp thường chỉ bao gồm một vài loại hệ thống con khác nhau trong các kết hợp và sắp xếp khác nhau.
* Nói cách khác, các hệ thống phức tạp có các mẫu phổ biến. Những mẫu này có thể liên quan đến việc tái sử dụng các thành phần nhỏ, chẳng hạn như các tế bào được tìm thấy trong cả hai thực vật và động vật hoặc các cấu trúc lớn hơn, chẳng hạn như hệ thống mạch máu, cũng được tìm thấy ở cả hai thực vật và động vật.
  1. Stable intermediate forms.
* Các hệ thống phức tạp sẻ phát triển từ các hệ thống đơn giản nhanh hơn nhiều nếu có là hình thức trung gian ổn định hơn nếu không có.
* Khi các hệ thống phát triển, các đối tượng đã từng được coi là phức tạp trở thành cơ sở các đối tượng mà trên đó các hệ thống phức tạp hơn được xây dựng.
* Chúng ta phải sử dụng chúng trong ngữ cảnh đầu tiên và sau đó cải thiện chúng theo thời gian khi chúng ta tìm hiểu thêm về hành vi thực sự của hệ thống.

1. Nền tảng của mô hình đối tượng.
   * Sử dụng lớp và đối tượng.
   * Mô hình đối tượng dùng cho thiết kế giao diện người dùng, cơ sở dữ liệu và thậm chí cả kiến trúc máy tính.
   * Phân thích và thiết kế hướng đối tượng đại diện cho một sự phát triển.
2. Các thành phần của mô hình đối tượng.

+Lớp

+Đối tượng

+Mối quan hệ giữa lớp và đối tượng

1. Đối tượng và quan hệ giữa các đối tượng.

+ Đối tượng là khái niệm cho phép mô tả các sự vật/thực thể trong thế giới thực.

+Các đối tượng duy trì mối quan hệ giữa chúng

+ Ví dụ: Nguyễn Văn A là một đối tượng

+ Các tính chất của đối tượng

Đối tượng = trạng thái + hành vi + định danh

• Trạng thái là các đặc tính của đối tượng tại một thời điểm

• Hành vi thể hiện các chức năng của đối tượng

• Định danh thể hiện sự tồn tại duy nhất của đối tượng

Trạng thái = tập hợp các thuộc tính

+Mỗi thuộc tính mô tả một đặc tính

+Tại một thời điểm cụ thể, các thuộc tính mang các giá trị trong miền xác định

+Ví dụ Một chiếc xe máy: màu xanh, 110 cm3, dream, 12000km,…

Hành vi = tập hợp các phương thức

Phương thức: là một thao tác hoặc được thực hiện bởi chính nó, hoặc thực hiện khi có yêu cầu từ môi trường (thông điệp từ đối tượng khác)

+Hành vi phụ thuộc vào trạng thái

+ Ví dụ một xe máy có các hành vi: khởi động, chạy, …

Quan hệ giữa các đối tượng:

Các đối tượng giao tiếp với nhau

Gửi các thông điệp (message) cho nhau

Các loại thông điệp

Hàm dựng (constructor)

Hàm hủy (destructor)

Hàm chọn lựa (get)

Hàm sửa đổi (set)

Các hàm chức năng khác

Giữa các đối tượng có mối liên kết (link) với nhau

1. Lớp và quan hệ giữa các lớp.

+ Lớp là khái niệm dùng để mô tả một tập hợp các đối tượng có cùng một cấu trúc, cùng hành vi và có cùng những mối quan hệ với các đối tượng khác

+Lớp = các thuộc tính + các phương thức

+Lớp là một bước trừu tượng hóa

-Tìm kiếm các điểm giống, bỏ qua các điểm khác nhau của đối tượng

- Trừu tượng hóa làm giảm độ phức tạp

+Quan hệ giữa các lớp: kết hợp  
+ Một kết hợp là một tập hợp các mối liên kết giữa các đối tượng

1. Xác định lớp và đối tượng.

Đối tượng là thể hiện (instance) của lớp

Giá trị là thể hiện của thuộc tính

Liên kết là thể hiện của kết hợp

Lớp → đối tượng

Thuộc tính → giá trị

Kết hợp → liên kết

1. Vẽ sơ đồ use case và viết mô tả, sơ đồ hoạt động, sơ đồ lớp, sơ đồ tuần tự.
   1. Hệ thống trắc nghiệm trực tuyến.
      * Admin quản trị danh sách user thi, đề thi, xuất kết quả điểm, quản lý các đợt thi của user...
      * User đăng nhập để thi, xem kết quả và in kết quả.
      * Mỗi user chỉ thi một lần.
   2. Hệ thống khảo sát việc làm.
      * User đăng nhập bằng thông tin các nhân.
      * Tìm kiếm thông tin khóa học, văn bằng.
      * Cập nhật phiếu khảo sát chi tiết.
      * Cập nhật thông tin việc làm từ lúc tốt nghiệp đến nay.
      * Admin: import dữ liệu danh sách khóa học, thông tin văn bằng.

**1. Khái niệm về Phân tích và thiết kế hướng đối tượng (Object Oriented Analysis and Design: OOAD)**

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD)