

## **I. Câu hỏi trắc nghiệm:**

**Câu hỏi 1:** Lớp nào dưới đây đại diện cho các đối tượng có dữ liệu và trạng thái cần quản lý trong hệ thống?

- A. Lớp biên
- B. Lớp điều khiển
- C. Lớp thực thể
- D. Lớp giao diện

**Câu hỏi 2:** Lớp biên trong hệ thống có vai trò gì?

- A. Xử lý dữ liệu
- B. Điều khiển luồng công việc
- C. Giao tiếp với người dùng hoặc hệ thống bên ngoài
- D. Lưu trữ dữ liệu

**Câu hỏi 3:** Quan hệ nào giữa các lớp thể hiện sự kế thừa?

- A. Association
- B. Aggregation
- C. Composition
- D. Inheritance

**Câu hỏi 4:** Sơ đồ lớp mô tả:

- A. Quan hệ giữa các đối tượng trong một luồng xử lý
- B. Các lớp và quan hệ giữa các lớp trong hệ thống
- C. Giao diện người dùng của hệ thống
- D. Thứ tự luồng xử lý giữa các đối tượng

**Câu hỏi 5:** Quan hệ Include giữa các use case được dùng khi:

- A. Một use case mở rộng một use case khác
- B. Một use case cần gọi một use case khác để hoàn thành chức năng
- C. Một use case kế thừa một use case khác
- D. Một use case được sử dụng bởi hệ thống bên ngoài

**Câu hỏi 6:** Scenario là gì?

- A. Một sơ đồ mô tả lớp trong hệ thống
- B. Một kịch bản mô tả cách hệ thống và người dùng tương tác
- C. Một sơ đồ mô tả các luồng dữ liệu
- D. Một tài liệu thiết kế giao diện

**Câu hỏi 7:** Quan hệ nào sau đây biểu diễn việc một lớp chứa một lớp khác nhưng lớp con vẫn có thể tồn tại độc lập?

- A. Aggregation
- B. Composition
- C. Association
- D. Inheritance

**Câu hỏi 8:** Sơ đồ tuần tự mô tả điều gì?

- A. Quan hệ giữa các lớp
- B. Thứ tự các thông điệp được trao đổi giữa các đối tượng
- C. Cấu trúc dữ liệu của hệ thống
- D. Các chức năng mà hệ thống cung cấp

**Câu hỏi 9:** Lớp điều khiển trong mô hình MVC tương ứng với thành phần nào?

- A. Model
- B. View
- C. Control
- D. Entity

**Câu hỏi 10:** Để biểu diễn quan hệ giữa các lớp, ta sử dụng sơ đồ nào?

- A. Sơ đồ tuần tự
- B. Sơ đồ cộng tác
- C. Sơ đồ lớp
- D. Sơ đồ use case

## **II. Câu hỏi trả lời ngắn:**

**Câu 1:** Lớp thực thể là gì?

- Lớp thực thể (Entity Class) là các lớp mô tả các thực thể trong hệ thống phần mềm, thường được ánh xạ trực tiếp đến bảng trong cơ sở dữ liệu.
- Chúng lưu trữ dữ liệu có trạng thái và tồn tại lâu dài trong hệ thống.

**Câu 2: Lớp điều khiển có vai trò gì trong hệ thống?**

- Lớp điều khiển (Control Class) đóng vai trò trung gian giữa lớp giao diện (Boundary Class) và lớp thực thể (Entity Class).
- Nó chịu trách nhiệm xử lý logic nghiệp vụ và điều phối luồng dữ liệu trong hệ thống.

**Câu 3: Scenario là gì?:**

- Scenario là tập hợp các bước mô tả chi tiết về cách người dùng tương tác với hệ thống để hoàn thành một nhiệm vụ cụ thể.
- Nó giúp mô tả cụ thể luồng tương tác giữa người dùng và hệ thống, hỗ trợ phát triển use case và kiểm thử phần mềm.

**Câu 4: Quan hệ Include giữa các use case là gì?**

- Quan hệ Include mô tả một use case cần gọi một use case khác để hoàn thành chức năng
- Use case chính **luôn luôn** gọi use case được Include, không thể hoạt động nếu không có nó.

**Câu 5: Mục đích của sơ đồ lớp là gì?:**

- Sơ đồ lớp (Class Diagram) giúp mô tả cấu trúc tĩnh của hệ thống phần mềm.
- Nó biểu diễn các lớp, thuộc tính, phương thức và mối quan hệ giữa các lớp để giúp thiết kế hệ thống dễ hiểu và bảo trì.

**Câu 6: Quan hệ Aggregation khác gì so với Composition?**

- **Aggregation (Tập hợp):** Một lớp chứa một lớp khác, nhưng lớp con **vẫn có thể tồn tại độc lập**.
- **Composition (Thành phần):** Một lớp chứa một lớp khác, nhưng lớp con **không thể tồn tại độc lập** nếu lớp cha bị xóa.

**Câu 7: Sơ đồ tuần tự là gì?**

- Sơ đồ tuần tự (Sequence Diagram) mô tả thứ tự các thông điệp được trao đổi giữa các đối tượng trong hệ thống theo thời gian.
- Nó giúp phân tích quy trình xử lý nghiệp vụ và kiểm tra luồng thực thi của hệ thống.

**Câu 8: Quan hệ Extend giữa các use case là gì?**

- Quan hệ Extend mô tả một use case mở rộng một use case khác trong **một số điều kiện cụ thể**.
- Use case mở rộng chỉ được thực thi khi điều kiện đặc biệt xảy ra, không bắt buộc lúc nào cũng thực hiện.

**Câu 9: Lớp biên có vai trò gì trong hệ thống?**

- Lớp biên (Boundary Class) là lớp giao tiếp giữa hệ thống và người dùng hoặc hệ thống bên ngoài.
- Nó xử lý đầu vào của người dùng, hiển thị thông tin, và có thể tương tác với API hoặc cơ sở dữ liệu.

**Câu 10: Sơ đồ cộng tác là gì?**

- Sơ đồ cộng tác (Collaboration Diagram) mô tả cách các đối tượng tương tác với nhau trong hệ thống.
- Nó tập trung vào **quan hệ giữa các đối tượng** thay vì trình tự thời gian như sơ đồ tuần tự

**III. Câu hỏi thảo luận nhóm:**

**Câu 1: Vai trò của từng loại lớp trong hệ thống (Thực thể, Biên, Điều khiển):**

- Lớp thực thể (Entity Class):
  - Chứa dữ liệu và trạng thái của hệ thống.
  - Được ánh xạ đến bảng trong cơ sở dữ liệu.
  - Không chứa logic nghiệp vụ.
- Lớp biên (Boundary Class):
  - Giao tiếp giữa hệ thống và người dùng/hệ thống bên ngoài.
  - Xử lý đầu vào và hiển thị dữ liệu.
  - Có thể tương tác với API hoặc cơ sở dữ liệu.
- Lớp điều khiển (Control Class):
  - Xử lý logic nghiệp vụ và điều phối luồng dữ liệu.
  - Đóng vai trò trung gian giữa lớp biên và lớp thực thể.

**Câu 2: So sánh Aggregation và Composition:**

Đặc điểm	Aggregation	Composition
Quan hệ	Có, nhưng không bắt buộc	Có và bắt buộc
Sự phụ thuộc	Lớp con có thể tồn tại độc lập	Lớp con không thể tồn tại nếu lớp cha bị xóa
Ví dụ	Một lớp “Công ty” chứa nhiều “Nhân viên” (Nhân viên có thể tồn tại nếu công ty bị xóa)	Một lớp “Xe hơi” chứa “Động cơ” (Nếu xe bị xóa, động cơ cũng bị xóa)

**Câu 3: Tầm quan trọng của sơ đồ lớp trong phân tích hệ thống:**

- Giúp biểu diễn cấu trúc hệ thống rõ ràng
- Làm tài liệu tham khảo cho lập trình viên
- Giúp xác định quan hệ giữa các lớp
- Tạo nền tảng cho các sơ đồ khác nhau như sơ đồ tuần tự và cộng tác.

**Câu 4: Phân biệt sơ đồ tuần tự và sơ đồ cộng tác:**

Tiêu chí	Sơ đồ tuần tự (Sequence Diagram)	Sơ đồ cộng tác (Collaboration Diagram)
Tập trung vào	Trình tự các thông điệp giữa các đối tượng	Quan hệ giữa các đối tượng
Biểu diễn	Các thông điệp theo thời gian	Cấu trúc giao tiếp giữa các đối tượng
Khi nào dùng	Khi cần phân tích luồng xử lý	Khi cần hiểu rõ sự tương tác giữa các đối tượng

**Câu 5: Vai trò của lớp điều khiển trong mô hình MVC:**

- Mô hình MVC giúp tách biệt giao diện, logic nghiệp vụ và dữ liệu
- Lớp điều khiển: Nhận yêu cầu từ View, Xử lý logic và tương tác với Model, Trả về kết quả về View

**Câu 6: Tại sao cần viết các scenario khi phân tích hệ thống:**

- Mô tả chi tiết các người dùng tương tác với hệ thống
- Hỗ trợ kiểm thử phần mềm
- Đảm bảo hệ thống đáp ứng đúng yêu cầu
- Giúp phát hiện các trường hợp ngoại lệ

**Câu 7: Làm thế nào để đảm bảo rằng các use case được trích đầy đủ và chính xác:**

- Phỏng vấn kỹ Stakeholders
- Phân tích kỹ yêu cầu hệ thống
- Sử dụng kỹ thuật Use case Diagram và user story
- Thực hiện kiểm tra chéo với các chuyên gia

**Câu 8: Mối quan hệ giữa use case case và scenario:**

- Use case mô tả chức năng hệ thống ở mức tổng quát
- Scenario mô tả chi tiết từng tình huống cụ thể của use case
- Một use case có thể có nhiều scenario

**Câu 9: Ưu điểm và nhược điểm của sơ đồ tuần tự trong thiết kế hệ thống:**

- Ưu điểm: dễ hiểu, thể hiện luồng dữ liệu rõ ràng, giúp phát hiện lỗi sớm
- Nhược điểm: khó bảo trì khi hệ thống phức tạp, không mô tả được toàn bộ cấu trúc hệ thống

**Câu 10: Cách cải thiện chất lượng scenario trong phân tích hệ thống:**

- Viết rõ ràng, dễ hiểu
- Bổ sung các trường hợp ngoại lệ
- Kiểm tra lại với stakeholder
- Sử dụng mô hình use case để xác minh

**IV. Câu hỏi tình huống:****Câu 1: Hệ thống quản lý thư viện có yêu cầu mới sau khi viết scenario, xử lý thế nào?**

- Cập nhật lại Use Case.

- Điều chỉnh Scenario để phản ánh thay đổi.
- Đảm bảo các sơ đồ liên quan cũng được cập nhật.

**Câu 2: Nhóm phát triển gặp khó khăn khi xác định lớp điều khiển, giải pháp?**

- Xác định các Use Case chính.
- Tìm các luồng nghiệp vụ chính.
- Phân tách các chức năng để xác định lớp điều khiển phù hợp.

**Câu 3: Sau khi hoàn thành sơ đồ lớp, khách hàng yêu cầu thêm chức năng mới, làm gì?**

- Phân tích yêu cầu mới có ảnh hưởng gì đến hệ thống.
- Cập nhật sơ đồ lớp và các sơ đồ liên quan.
- Kiểm tra lại toàn bộ hệ thống để đảm bảo tính nhất quán.

**Câu 4: Xử lý khi có nhiều ngoại lệ trong Use Case "Đăng ký khóa học"**

- Phân loại ngoại lệ (Lỗi nhập liệu, Lỗi hệ thống, Lỗi logic).
- Viết Scenario riêng cho từng ngoại lệ.
- Cập nhật sơ đồ lớp và sơ đồ tuần tự nếu cần.

**Câu 5: Một số đối tượng trong sơ đồ tuần tự không có vai trò rõ ràng, làm gì?**

- Xác định lại luồng nghiệp vụ.
- Loại bỏ hoặc điều chỉnh các đối tượng không cần thiết.
- Đảm bảo mỗi đối tượng có chức năng cụ thể.

**Câu 6: Quan hệ giữa các lớp trong sơ đồ lớp bị sai, sửa thế nào?**

- Kiểm tra lại yêu cầu hệ thống.
- Sử dụng mô hình hóa để xác định lại quan hệ chính xác.
- Cập nhật sơ đồ lớp và kiểm tra tính nhất quán.

**Câu 7: Gặp khó khăn khi mô tả quan hệ giữa các Use Case, giải pháp?**

- Sử dụng sơ đồ Use Case để xác định rõ các mối quan hệ.
- Kiểm tra các quan hệ Include và Extend.
- Hỏi ý kiến stakeholders để làm rõ hơn.

**Câu 8: Xác định các lớp biên cho hệ thống quản lý bán hàng**

- **CustomerInterface:** Xử lý giao diện khách hàng.
- **AdminInterface:** Quản lý sản phẩm, đơn hàng.
- **PaymentInterface:** Xử lý thanh toán.

**Câu 9: Khách hàng yêu cầu thêm chức năng mới sau khi hoàn thiện scenario, xử lý thế nào?**

- Phân tích mức độ ảnh hưởng của chức năng mới.
- Cập nhật lại các Scenario liên quan.
- Kiểm tra lại sơ đồ hệ thống để đảm bảo tính nhất quán.

**Câu 10: Sơ đồ cộng tác có một số đối tượng không tương tác đúng, giải quyết thế nào?**

- Xem lại kịch bản nghiệp vụ để xác định đúng vai trò của đối tượng.
- Sửa đổi các thông điệp giữa các đối tượng.
- Kiểm tra lại sơ đồ tuần tự để đảm bảo tương tác đúng.