Họ và tên: Lâm Thành Trung  
MSSV: N18DCAT096  
Nhóm 10  
Bài tập tuần 4 – Nhập môn công nghệ phần mềm – D22CQAT01-N  
  
**Câu hỏi chương 4:**

**Câu hỏi trắc nghiệm:**   
  
Câu hỏi 1: Mục tiêu chính của kiểm thử phần mềm là gì?

Đáp án: B. Đảm bảo phần mềm đáp ứng đúng yêu cầu

Giải thích: Mục tiêu quan trọng nhất của kiểm thử phần mềm là xác minh rằng phần mềm hoạt động như mong đợi và đáp ứng các yêu cầu đã được xác định. Các lựa chọn khác như loại bỏ tất cả các lỗi là bất khả thi trong thực tế.

Câu hỏi 2: Nhóm nào chịu trách nhiệm đảm bảo chất lượng phần mềm trong dự án?

Đáp án: C. Nhóm SQA (Software Quality Assurance)

Giải thích: Nhóm SQA chịu trách nhiệm thiết lập và thực hiện các quy trình đảm bảo chất lượng trong suốt vòng đời phát triển phần mềm.

Câu hỏi 3: Kiểm thử các sản phẩm phi thực thi bao gồm hoạt động nào dưới đây?

Đáp án: B. Walkthrough và review tài liệu

Giải thích: Các sản phẩm phi thực thi như tài liệu thiết kế, yêu cầu, v.v., được kiểm tra thông qua các hoạt động như walkthrough và review để tìm ra các lỗi và sai sót.

Câu hỏi 4: Lập kế hoạch trong tiến trình phát triển phần mềm nhằm mục đích gì?

Đáp án: A. Xác định phạm vi công việc và ước tính thời gian hoàn thành

Giải thích: Lập kế hoạch giúp xác định rõ ràng các công việc cần thực hiện, nguồn lực cần thiết và thời gian dự kiến để hoàn thành dự án.

Câu hỏi 5: Loại kiểm thử nào sau đây tập trung vào việc phát hiện các lỗi chức năng của phần mềm?

Đáp án: B. Kiểm thử chức năng

Giải thích: Kiểm thử chức năng nhằm mục đích xác minh rằng các chức năng của phần mềm hoạt động đúng theo yêu cầu.

Câu hỏi 6: Quản lý phiên bản tài liệu có mục tiêu gì?

Đáp án: A. Đảm bảo tài liệu luôn được cập nhật và có thể truy xuất phiên bản cũ khi cần

Giải thích: Quản lý phiên bản tài liệu giúp theo dõi các thay đổi và đảm bảo tính nhất quán của tài liệu trong suốt quá trình phát triển.

Câu hỏi 7: Nhóm SQA có vai trò gì trong kiểm thử phần mềm?

Đáp án: B. Đánh giá và đảm bảo quy trình phát triển phần mềm tuân thủ tiêu chuẩn chất lượng

Giải thích: Nhóm SQA tập trung vào việc đảm bảo rằng các quy trình được tuân thủ và sản phẩm đáp ứng các tiêu chuẩn chất lượng.

Câu hỏi 8: Loại kiểm thử nào thường được thực hiện cuối cùng trước khi phần mềm được bàn giao cho khách hàng?

Đáp án: D. Kiểm thử chấp nhận (Acceptance Testing)

Giải thích: Kiểm thử chấp nhận được thực hiện bởi khách hàng hoặc người dùng cuối để xác minh rằng phần mềm đáp ứng yêu cầu của họ trước khi chấp nhận.

Câu hỏi 9: Đâu là một trong những công cụ phổ biến dùng để quản lý phiên bản tài liệu?

Đáp án: B. Git

Giải thích: Git là một hệ thống kiểm soát phiên bản phân tán phổ biến được sử dụng rộng rãi để quản lý mã nguồn và tài liệu.

Câu hỏi 10: Hoạt động lập tài liệu trong mỗi pha phát triển phần mềm nhằm mục đích gì?

Đáp án: B. Hỗ trợ quá trình bảo trì và nâng cấp phần mềm sau khi triển khai

Giải thích: Tài liệu cung cấp thông tin cần thiết để hiểu và sửa đổi phần mềm trong quá trình bảo trì và nâng cấp.

**Câu hỏi ngắn:**1. Nhóm SQA là gì và vai trò của nhóm này trong phát triển phần mềm?

* SQA (Software Quality Assurance) là viết tắt của Đảm bảo chất lượng phần mềm. Đây là một nhóm hoặc bộ phận trong tổ chức phát triển phần mềm chịu trách nhiệm thiết lập và giám sát việc thực hiện các quy trình và tiêu chuẩn chất lượng.
* Vai trò của nhóm SQA:
  + Xây dựng và duy trì các quy trình và tiêu chuẩn chất lượng.
  + Đánh giá và kiểm tra các sản phẩm phần mềm và quy trình phát triển để đảm bảo chúng tuân thủ các tiêu chuẩn đã định.
  + Xác định và đề xuất các cải tiến quy trình để nâng cao chất lượng phần mềm.
  + Thực hiện các cuộc đánh giá và kiểm toán để xác định sự tuân thủ quy trình.
  + Cung cấp báo cáo về chất lượng phần mềm và các vấn đề liên quan.

2. Kiểm thử đơn vị là gì?

* Kiểm thử đơn vị (Unit Testing) là một phương pháp kiểm thử phần mềm trong đó các đơn vị mã nguồn riêng lẻ (như hàm, phương thức) được kiểm tra độc lập để xác minh chúng hoạt động đúng như mong đợi.
* Mục tiêu của kiểm thử đơn vị là phát hiện sớm các lỗi trong mã nguồn và đảm bảo rằng các thành phần riêng lẻ của phần mềm hoạt động chính xác.

3. Mục tiêu chính của kiểm thử chấp nhận là gì?

* Mục tiêu chính của kiểm thử chấp nhận (Acceptance Testing) là xác minh rằng phần mềm đáp ứng các yêu cầu của người dùng cuối hoặc khách hàng.
* Kiểm thử chấp nhận thường được thực hiện bởi người dùng cuối hoặc đại diện của họ và tập trung vào việc đánh giá tính khả dụng và tính phù hợp của phần mềm trong môi trường thực tế.

4. Các hoạt động chính trong kiểm thử sản phẩm phi thực thi là gì?

* Các hoạt động chính trong kiểm thử sản phẩm phi thực thi bao gồm:
  + Review tài liệu: Đánh giá các tài liệu như yêu cầu, thiết kế, và tài liệu hướng dẫn để tìm lỗi và sự không nhất quán.
  + Walkthrough: Một quá trình đánh giá trong đó các thành viên nhóm xem xét tài liệu hoặc mã nguồn và thảo luận về các vấn đề tiềm ẩn.
  + Inspection: Một quá trình đánh giá chính thức hơn, trong đó các chuyên gia kiểm tra tài liệu hoặc mã nguồn theo các tiêu chuẩn cụ thể.

5. Tại sao việc lập tài liệu cho mỗi pha phát triển phần mềm lại quan trọng?

* Việc lập tài liệu cho mỗi pha phát triển phần mềm rất quan trọng vì:
  + Nó giúp đảm bảo rằng tất cả các thành viên nhóm đều có cùng hiểu biết về dự án.
  + Nó cung cấp một hồ sơ về các quyết định thiết kế và yêu cầu.
  + Nó hỗ trợ quá trình bảo trì và nâng cấp phần mềm sau khi triển khai.
  + Nó giúp cho việc đào tạo người dùng và các thành viên nhóm mới.
  + Nó là bằng chứng để đảm bảo tính minh bạch của dự án.

6. Quản lý phiên bản tài liệu là gì?

* Quản lý phiên bản tài liệu (Document Version Control) là quá trình theo dõi và quản lý các thay đổi đối với tài liệu trong suốt vòng đời của chúng.
* Mục tiêu của quản lý phiên bản tài liệu là:
  + Đảm bảo rằng luôn có một phiên bản chính xác và cập nhật của tài liệu.
  + Cho phép truy xuất các phiên bản cũ hơn của tài liệu khi cần thiết.
  + Ngăn ngừa sự mất mát hoặc ghi đè dữ liệu.
  + Giúp nhiều người có thể cùng làm việc trên một tài liệu mà không gây ra xung đột.

7. Các loại kiểm thử chính trong kiểm thử sản phẩm thực thi là gì?

* Các loại kiểm thử chính trong kiểm thử sản phẩm thực thi bao gồm:
  + Kiểm thử đơn vị (Unit Testing): Kiểm tra các đơn vị mã nguồn riêng lẻ.
  + Kiểm thử tích hợp (Integration Testing): Kiểm tra sự tương tác giữa các thành phần phần mềm.
  + Kiểm thử hệ thống (System Testing): Kiểm tra toàn bộ hệ thống để đảm bảo nó đáp ứng các yêu cầu.
  + Kiểm thử chấp nhận (Acceptance Testing): Kiểm tra phần mềm từ quan điểm của người dùng cuối.
  + Kiểm thử hiệu năng (Performance Testing): Kiểm tra hiệu suất của phần mềm dưới các điều kiện tải khác nhau.
  + Kiểm thử bảo mật (Security Testing): Kiểm tra các lỗ hổng bảo mật của phần mềm.

8. Kiểm thử tích hợp là gì?

* Kiểm thử tích hợp (Integration Testing) là quá trình kiểm tra sự tương tác và giao tiếp giữa các thành phần khác nhau của phần mềm.
* Mục tiêu của kiểm thử tích hợp là đảm bảo rằng các thành phần này hoạt động cùng nhau một cách chính xác.

9. Hoạt động lập kế hoạch cho các pha phát triển phần mềm bao gồm những gì?

* Hoạt động lập kế hoạch cho các pha phát triển phần mềm bao gồm:
  + Xác định phạm vi công việc.
  + Ước tính thời gian và nguồn lực cần thiết.
  + Lập lịch trình và phân công công việc.
  + Xác định các rủi ro và kế hoạch ứng phó.
  + Lựa chọn các công cụ và công nghệ phù hợp.
  + Xác định các tiêu chuẩn và quy trình chất lượng.

10. Làm tài liệu kiểm thử bao gồm những gì?

* Làm tài liệu kiểm thử bao gồm:
  + Kế hoạch kiểm thử (Test Plan): Mô tả phạm vi, mục tiêu, và chiến lược kiểm thử.
  + Kịch bản kiểm thử (Test Scenarios): Mô tả các tình huống kiểm thử cụ thể.
  + Trường hợp kiểm thử (Test Cases): Mô tả chi tiết các bước thực hiện và kết quả mong đợi cho từng tình huống kiểm thử.
  + Dữ liệu kiểm thử (Test Data): Dữ liệu được sử dụng để thực hiện các trường hợp kiểm thử.
  + Báo cáo kiểm thử (Test Reports): Báo cáo kết quả kiểm thử và các vấn đề được phát hiện.
  + Ma trận truy vết yêu cầu (Requirement Traceability Matrix): Đảm bảo rằng tất cả các yêu cầu đều được kiểm thử.

**Câu hỏi thảo luận nhóm:**1. Vai trò của nhóm SQA trong việc đảm bảo chất lượng phần mềm là gì?

* Nhóm SQA (Software Quality Assurance - Đảm bảo chất lượng phần mềm) đóng vai trò then chốt trong việc duy trì và nâng cao chất lượng phần mềm.
* Vai trò chính của họ bao gồm:
  + Xây dựng và duy trì các quy trình và tiêu chuẩn chất lượng.
  + Đánh giá và kiểm tra các sản phẩm phần mềm để đảm bảo chúng tuân thủ các tiêu chuẩn đã định.
  + Xác định và đề xuất các cải tiến quy trình để nâng cao chất lượng.
  + Thực hiện các cuộc đánh giá và kiểm toán để xác định sự tuân thủ quy trình.
  + Cung cấp báo cáo về chất lượng phần mềm và các vấn đề liên quan.

2. Thảo luận về sự khác nhau giữa kiểm thử đơn vị và kiểm thử tích hợp.

* Kiểm thử đơn vị (Unit Testing) tập trung vào việc kiểm tra từng đơn vị mã nguồn riêng lẻ (như hàm, phương thức) để đảm bảo chúng hoạt động đúng như mong đợi. Mục tiêu là phát hiện sớm các lỗi trong mã nguồn.
* Kiểm thử tích hợp (Integration Testing) tập trung vào việc kiểm tra sự tương tác và giao tiếp giữa các thành phần khác nhau của phần mềm. Mục tiêu là đảm bảo rằng các thành phần này hoạt động cùng nhau một cách chính xác.
* Nói ngắn gọn, kiểm thử đơn vị kiểm tra các bộ phận riêng lẻ, còn kiểm thử tích hợp kiểm tra cách các bộ phận đó hoạt động cùng nhau.

3. Tại sao việc lập tài liệu kiểm thử lại quan trọng trong mỗi dự án phần mềm?

* Lập tài liệu kiểm thử rất quan trọng vì:
  + Nó cung cấp một hồ sơ chi tiết về các hoạt động kiểm thử, giúp đảm bảo tính minh bạch và khả năng tái lập.
  + Nó giúp các thành viên nhóm hiểu rõ về phạm vi và mục tiêu kiểm thử.
  + Nó hỗ trợ việc theo dõi tiến độ và đánh giá kết quả kiểm thử.
  + Nó giúp cho việc bảo trì và nâng cấp phần mềm sau này.

4. Thảo luận về các thách thức khi lập kế hoạch cho các pha phát triển phần mềm.

* Các thách thức khi lập kế hoạch bao gồm:
  + Ước tính thời gian và nguồn lực không chính xác.
  + Thay đổi yêu cầu trong quá trình phát triển.
  + Rủi ro kỹ thuật và công nghệ.
  + Sự không chắc chắn về các yếu tố bên ngoài.
  + Khó khăn trong việc phối hợp giữa các thành viên nhóm.

5. Quản lý phiên bản tài liệu có ảnh hưởng như thế nào đến quá trình bảo trì phần mềm?

* Quản lý phiên bản tài liệu giúp đảm bảo rằng luôn có một phiên bản chính xác và cập nhật của tài liệu.
* Điều này rất quan trọng trong quá trình bảo trì, vì nó giúp các nhà phát triển hiểu rõ về mã nguồn và các thay đổi đã được thực hiện.
* Nó cũng giúp cho việc truy xuất các phiên bản cũ hơn của tài liệu khi cần thiết.

6. So sánh giữa kiểm thử sản phẩm phi thực thi và kiểm thử sản phẩm thực thi.

* Kiểm thử sản phẩm phi thực thi tập trung vào việc đánh giá các tài liệu và sản phẩm không phải là mã nguồn thực thi, như tài liệu yêu cầu, thiết kế.
* Kiểm thử sản phẩm thực thi tập trung vào việc kiểm tra mã nguồn thực thi, như kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thống.
* Nói ngắn gọn, kiểm thử phi thực thi kiểm tra "giấy tờ", còn kiểm thử thực thi kiểm tra "phần mềm chạy".

7. Thảo luận về cách cải thiện quy trình lập kế hoạch để giảm thiểu rủi ro trong dự án phần mềm.

* Để cải thiện quy trình lập kế hoạch, cần:
  + Thu thập yêu cầu chi tiết và rõ ràng.
  + Ước tính thời gian và nguồn lực một cách cẩn thận.
  + Xác định và đánh giá rủi ro một cách kỹ lưỡng.
  + Lập kế hoạch dự phòng cho các rủi ro có thể xảy ra.
  + Sử dụng các công cụ và kỹ thuật lập kế hoạch hiệu quả.

8. Đề xuất các công cụ hỗ trợ việc lập tài liệu kiểm thử và quản lý phiên bản.

* Các công cụ hỗ trợ lập tài liệu kiểm thử:
  + TestRail
  + Zephyr
  + Xray
* Các công cụ hỗ trợ quản lý phiên bản:
  + Git
  + SVN

9. Tại sao kiểm thử chấp nhận lại là một giai đoạn quan trọng trong phát triển phần mềm?

* Kiểm thử chấp nhận là giai đoạn quan trọng vì:
  + Nó xác minh rằng phần mềm đáp ứng các yêu cầu của người dùng cuối.
  + Nó giúp đảm bảo rằng phần mềm sẵn sàng để triển khai.
  + Nó giúp tăng cường sự tin tưởng của khách hàng vào sản phẩm.

10. Thảo luận về các phương pháp hiệu quả để quản lý chất lượng phần mềm trong các dự án lớn.

* Các phương pháp hiệu quả bao gồm:
  + Áp dụng các quy trình và tiêu chuẩn chất lượng.
  + Thực hiện kiểm thử tự động.
  + Sử dụng các công cụ quản lý chất lượng.
  + Đào tạo và nâng cao năng lực cho các thành viên nhóm.
  + Thực hiện đánh giá và cải tiến quy trình liên tục.

**Câu hỏi tình huống:**1. Một dự án phần mềm đã hoàn thành và sắp bàn giao cho khách hàng, nhưng khách hàng yêu cầu kiểm tra lại toàn bộ tài liệu yêu cầu và thiết kế. Đội phát triển nên xử lý thế nào?

* Thái độ chuyên nghiệp: Đội phát triển cần thể hiện sự chuyên nghiệp và sẵn sàng hợp tác.
* Đánh giá yêu cầu: Xác định rõ phạm vi và mục đích của yêu cầu kiểm tra.
* Lập kế hoạch: Lập kế hoạch chi tiết cho việc kiểm tra tài liệu, bao gồm thời gian, nguồn lực và phương pháp.
* Thực hiện kiểm tra: Tiến hành kiểm tra tài liệu một cách cẩn thận và kỹ lưỡng.
* Báo cáo kết quả: Báo cáo kết quả kiểm tra cho khách hàng một cách rõ ràng và minh bạch.
* Thảo luận và giải quyết: Thảo luận với khách hàng về các vấn đề phát hiện và đưa ra giải pháp phù hợp.

2. Trong quá trình kiểm thử hệ thống, nhóm phát triển phát hiện một lỗi nghiêm trọng nhưng thời hạn bàn giao đang đến gần. Bạn sẽ xử lý tình huống này như thế nào?

* Đánh giá mức độ nghiêm trọng: Xác định mức độ ảnh hưởng của lỗi đến chức năng và hiệu suất của phần mềm.
* Ưu tiên sửa lỗi: Ưu tiên sửa lỗi nghiêm trọng trước các lỗi khác.
* Thông báo cho khách hàng: Thông báo cho khách hàng về tình trạng lỗi và kế hoạch khắc phục.
* Đánh giá lại thời hạn bàn giao: Đánh giá lại thời hạn bàn giao và thảo luận với khách hàng về khả năng gia hạn.
* Tập trung nguồn lực: Tập trung nguồn lực để sửa lỗi một cách nhanh chóng và hiệu quả.
* Kiểm tra lại: Kiểm tra lại kỹ lưỡng sau khi sửa lỗi để đảm bảo không còn vấn đề.

3. Một nhóm phát triển gặp khó khăn trong việc quản lý phiên bản tài liệu do tài liệu liên tục thay đổi. Hãy đề xuất giải pháp.

* Sử dụng hệ thống quản lý phiên bản: Áp dụng các công cụ quản lý phiên bản như Git, SVN để theo dõi và quản lý các thay đổi.
* Thiết lập quy trình quản lý phiên bản: Xây dựng quy trình rõ ràng về việc cập nhật, kiểm tra và phê duyệt các phiên bản tài liệu.
* Phân quyền truy cập: Phân quyền truy cập tài liệu cho các thành viên nhóm phù hợp.
* Giao tiếp hiệu quả: Đảm bảo sự giao tiếp hiệu quả giữa các thành viên nhóm về các thay đổi tài liệu.
* Đào tạo về quản lý phiên bản: Đào tạo cho các thành viên nhóm về cách sử dụng hệ thống quản lý phiên bản và quy trình quản lý.

4. Dự án phát triển phần mềm gặp vấn đề khi khách hàng yêu cầu thay đổi lớn trong pha cài đặt. Đội phát triển nên xử lý thế nào?

* Đánh giá tác động: Đánh giá tác động của thay đổi đến thời gian, chi phí và nguồn lực của dự án.
* Thảo luận với khách hàng: Thảo luận với khách hàng về khả năng thực hiện thay đổi và các chi phí phát sinh.
* Lập kế hoạch thay đổi: Lập kế hoạch chi tiết cho việc thực hiện thay đổi.
* Ưu tiên các thay đổi quan trọng: Ưu tiên thực hiện các thay đổi quan trọng nhất.
* Kiểm tra kỹ lưỡng: Kiểm tra kỹ lưỡng sau khi thực hiện thay đổi để đảm bảo không ảnh hưởng đến các chức năng khác.

5. Nhóm kiểm thử phát hiện nhiều lỗi chức năng trong phần mềm. Tuy nhiên, nhóm phát triển lại cho rằng đây không phải lỗi mà là tính năng. Là trưởng dự án, bạn sẽ làm gì?

* Lắng nghe cả hai bên: Lắng nghe ý kiến của cả nhóm kiểm thử và nhóm phát triển.
* Xem xét tài liệu yêu cầu: Xem xét lại tài liệu yêu cầu để xác định rõ chức năng mong muốn.
* Thảo luận và thống nhất: Tổ chức cuộc họp để thảo luận và thống nhất về định nghĩa lỗi và tính năng.
* Đưa ra quyết định: Đưa ra quyết định cuối cùng dựa trên tài liệu yêu cầu và ý kiến của cả hai bên.
* Ghi nhận và cải tiến: Ghi nhận lại các vấn đề phát sinh và cải tiến quy trình để tránh xảy ra tình huống tương tự.

6. Khách hàng yêu cầu bổ sung một tính năng mới khi phần mềm đã hoàn thành pha kiểm thử tích hợp. Đội phát triển nên làm gì?

* Đánh giá tác động: Đánh giá tác động của tính năng mới đến thời gian, chi phí và nguồn lực của dự án.
* Thảo luận với khách hàng: Thảo luận với khách hàng về khả năng thực hiện tính năng mới và các chi phí phát sinh.
* Lập kế hoạch bổ sung: Lập kế hoạch chi tiết cho việc bổ sung tính năng mới.
* Ưu tiên các tính năng quan trọng: Ưu tiên thực hiện các tính năng quan trọng nhất.
* Kiểm tra kỹ lưỡng: Kiểm tra kỹ lưỡng sau khi bổ sung tính năng để đảm bảo không ảnh hưởng đến các chức năng khác.

7. Một công ty phát triển phần mềm nhỏ muốn xây dựng nhóm SQA nhưng gặp khó khăn về ngân sách. Hãy đề xuất giải pháp.

* Tận dụng nguồn lực nội bộ: Đào tạo và phân công trách nhiệm SQA cho các thành viên nhóm hiện tại.
* Sử dụng các công cụ SQA miễn phí hoặc mã nguồn mở: Tìm kiếm và sử dụng các công cụ SQA phù hợp với ngân sách.
* Tập trung vào các hoạt động SQA quan trọng: Ưu tiên thực hiện các hoạt động SQA quan trọng nhất, như kiểm thử đơn vị và kiểm thử tích hợp.
* Xây dựng quy trình SQA đơn giản: Xây dựng quy trình SQA đơn giản và dễ thực hiện.
* Hợp tác với các chuyên gia SQA bên ngoài: Hợp tác với các chuyên gia SQA bên ngoài để được tư vấn và hỗ trợ.

8. Trong quá trình làm tài liệu kiểm thử, nhóm phát triển không thống nhất được về nội dung cần đưa vào tài liệu. Là trưởng nhóm, bạn sẽ giải quyết vấn đề này như thế nào?

* Tổ chức cuộc họp: Tổ chức cuộc họp để các thành viên nhóm thảo luận và thống nhất về nội dung tài liệu.
* Xác định mục tiêu: Xác định rõ mục tiêu của tài liệu kiểm thử và nội dung cần thiết để đạt được mục tiêu đó.
* Tham khảo các tài liệu mẫu: Tham khảo các tài liệu kiểm thử mẫu để có thêm ý tưởng.
* Đưa ra quyết định: Đưa ra quyết định cuối cùng về nội dung tài liệu dựa trên ý kiến của các thành viên nhóm và mục tiêu của dự án.
* Ghi nhận và cải tiến: Ghi nhận lại các vấn đề phát sinh và cải tiến quy trình làm tài liệu để tránh xảy ra tình huống tương tự.

9. Dự án phát triển phần mềm cho một ngân hàng yêu cầu bảo mật cao. Đề xuất cách lập kế hoạch kiểm thử cho dự án này.

* Ưu tiên kiểm thử bảo mật: Ưu tiên thực hiện kiểm thử bảo mật ngay từ giai đoạn đầu của dự án.
* Sử dụng các công cụ kiểm thử bảo mật chuyên dụng: Sử dụng các công cụ kiểm thử bảo mật như OWASP ZAP, Burp Suite.
* Thực hiện kiểm thử xâm nhập: Thực hiện kiểm thử xâm nhập để đánh giá khả năng chống lại các cuộc tấn công.
* Kiểm tra các tiêu chuẩn bảo mật: Kiểm tra sự tuân thủ các tiêu chuẩn bảo mật như PCI DSS, ISO 27001.
* Đào tạo về bảo mật: Đào tạo cho các thành viên nhóm về các nguyên tắc và kỹ thuật bảo mật.

10. Sau khi triển khai phần mềm, khách hàng phát hiện ra một số lỗi bảo mật nghiêm trọng. Đội phát triển cần xử lý ra sao để khắc phục vấn đề và lấy lại niềm tin từ khách hàng?

* Xác nhận và đánh giá: Xác nhận các lỗi bảo mật và đánh giá mức độ nghiêm trọng.
* Ưu tiên khắc phục: Ưu tiên khắc phục các lỗi bảo mật một cách nhanh chóng và hiệu quả.
* Thông báo cho khách hàng: Thông báo cho khách hàng về tình trạng lỗi và kế hoạch khắc phục.
* Kiểm tra lại kỹ lưỡng: Kiểm tra lại kỹ lưỡng sau khi khắc phục để đảm bảo không còn vấn đề.
* Báo cáo chi tiết: Báo cáo chi tiết cho khách hàng về nguyên nhân, cách khắc phục và các biện pháp phòng ngừa.
* Đề xuất các biện pháp bảo mật bổ sung: Đề xuất các biện pháp bảo mật bổ sung để tăng cường bảo vệ cho phần mềm.
* Cam kết cải tiến: Cam kết cải tiến quy trình phát triển và kiểm thử để tránh xảy ra lỗi tương tự.

**Câu hỏi chương 5:**

**Câu hỏi trắc nghiệm:**

**Câu hỏi 1: Kỹ thuật nào sau đây thường được sử dụng để thu thập yêu cầu khách hàng?**

**Đáp án: B. Phỏng vấn khách hàng**

Giải thích: Phỏng vấn khách hàng là một trong những phương pháp trực tiếp và hiệu quả nhất để hiểu rõ nhu cầu và mong muốn của họ.

**Câu hỏi 2: Trong bước tổng hợp kết quả yêu cầu, việc nào sau đây là cần thiết?**

**Đáp án: B. Phân loại yêu cầu và loại bỏ thông tin trùng lặp**

Giải thích: Bước này giúp tổ chức thông tin thu thập được một cách có hệ thống, loại bỏ sự dư thừa và đảm bảo tính nhất quán.

**Câu hỏi 3: Use case mô tả:**

**Đáp án: B. Cách người dùng tương tác với hệ thống**

Giải thích: Use case thể hiện các tình huống sử dụng hệ thống từ góc độ người dùng, giúp xác định rõ các chức năng cần thiết.

**Câu hỏi 4: Tại sao cần xây dựng danh sách từ khóa chuyên môn khi tìm hiểu lĩnh vực ứng dụng?**

**Đáp án: C. Để đảm bảo đội phát triển và khách hàng hiểu rõ các thuật ngữ**

Giải thích: Điều này giúp tránh hiểu lầm và đảm bảo sự giao tiếp hiệu quả giữa các bên liên quan.

**Câu hỏi 5: Quan hệ nào sau đây trong use case mô tả một use case có thể mở rộng thêm chức năng trong một số điều kiện nhất định?**

**Đáp án: B. Extend**

Giải thích: Quan hệ "extend" cho phép một use case mở rộng hành vi của một use case khác dưới các điều kiện cụ thể.

**Câu hỏi 6: Khi mô tả yêu cầu bằng ngôn ngữ tự nhiên, điều quan trọng nhất là gì?**

**Đáp án: C. Viết rõ ràng, dễ hiểu và tránh từ đa nghĩa**

Giải thích: Ngôn ngữ tự nhiên cần được sử dụng một cách chính xác để tránh gây hiểu lầm và đảm bảo tính chính xác của yêu cầu.

**Câu hỏi 7: Mục tiêu chính của việc trích các use case là gì?**

**Đáp án: B. Mô tả các chức năng mà hệ thống cung cấp cho người dùng**

Giải thích: Use case giúp xác định rõ các tương tác giữa người dùng và hệ thống, từ đó xác định các chức năng cần thiết.

**Câu hỏi 8: Quan hệ nào giữa các use case thể hiện việc một use case phải gọi một use case khác để thực hiện chức năng đầy đủ?**

**Đáp án: A. Include**

Giải thích: Quan hệ "include" cho biết một use case bắt buộc phải sử dụng một use case khác để hoàn thành chức năng của mình.

**Câu hỏi 9: Hoạt động nào sau đây là bước đầu tiên trong việc xây dựng mô hình nghiệp vụ?**

**Đáp án: B. Trích các use case**

Giải thích: Trích các use case là bước đầu tiên để hiểu rõ các quy trình nghiệp vụ và tương tác của người dùng với hệ thống.

**Câu hỏi 10: Một yêu cầu chức năng là gì?**

**Đáp án: A. Một yêu cầu mô tả cách hệ thống xử lý dữ liệu**

Giải thích: Yêu cầu chức năng mô tả các hành động cụ thể mà hệ thống cần thực hiện để đáp ứng nhu cầu của người dùng.

**Câu hỏi trả lời ngắn:**1. Kỹ thuật phỏng vấn khách hàng là gì?

* Kỹ thuật phỏng vấn khách hàng là quá trình thu thập thông tin về yêu cầu, mong muốn và nhu cầu của khách hàng thông qua các cuộc trò chuyện trực tiếp.
* Mục tiêu là hiểu rõ nhu cầu của khách hàng, xác định các yêu cầu chức năng và phi chức năng của hệ thống.
* Có nhiều hình thức phỏng vấn: phỏng vấn có cấu trúc, phỏng vấn bán cấu trúc và phỏng vấn không cấu trúc.

2. Tại sao cần tổng hợp kết quả yêu cầu?

* Tổng hợp kết quả yêu cầu giúp:
  + Tổ chức và phân loại các yêu cầu thu thập được.
  + Loại bỏ các yêu cầu trùng lặp hoặc mâu thuẫn.
  + Đảm bảo tính nhất quán và đầy đủ của yêu cầu.
  + Tạo cơ sở cho việc xây dựng mô hình nghiệp vụ và thiết kế hệ thống.

3. Use case là gì?

* Use case là một kỹ thuật mô tả các tương tác giữa người dùng (actor) và hệ thống để đạt được một mục tiêu cụ thể.
* Nó thể hiện các tình huống sử dụng hệ thống từ góc độ người dùng, giúp xác định rõ các chức năng cần thiết.
* Use case bao gồm các bước thực hiện, điều kiện trước và điều kiện sau.

4. Mục tiêu của việc xây dựng danh sách từ khóa chuyên môn là gì?

* Mục tiêu là đảm bảo sự hiểu biết chung giữa đội phát triển và khách hàng về các thuật ngữ chuyên môn trong lĩnh vực ứng dụng.
* Điều này giúp tránh hiểu lầm và đảm bảo sự giao tiếp hiệu quả trong suốt quá trình phát triển.

5. Quan hệ Include giữa các use case là gì?

* Quan hệ Include thể hiện việc một use case bắt buộc phải sử dụng một use case khác để hoàn thành chức năng của mình.
* Use case được include thường là các chức năng chung, được sử dụng bởi nhiều use case khác.

6. Quan hệ Extend giữa các use case là gì?

* Quan hệ Extend cho phép một use case mở rộng hành vi của một use case khác dưới các điều kiện cụ thể.
* Use case mở rộng (extension) chỉ được thực hiện khi đáp ứng các điều kiện nhất định.

7. Yêu cầu phi chức năng là gì?

* Yêu cầu phi chức năng mô tả các thuộc tính chất lượng của hệ thống, như hiệu suất, bảo mật, khả năng mở rộng, khả năng bảo trì, v.v.
* Chúng không mô tả các chức năng cụ thể của hệ thống, nhưng lại ảnh hưởng đến trải nghiệm người dùng và sự thành công của dự án.

8. Mô hình nghiệp vụ là gì?

* Mô hình nghiệp vụ là một biểu diễn trực quan của các quy trình nghiệp vụ, luồng thông tin và tương tác giữa các bên liên quan trong một tổ chức.
* Nó giúp hiểu rõ cách thức hoạt động của tổ chức và xác định các cơ hội cải tiến.

9. Điều kiện trước và điều kiện sau của use case là gì?

* Điều kiện trước (precondition) là các điều kiện phải được đáp ứng trước khi use case có thể được thực hiện.
* Điều kiện sau (postcondition) là các điều kiện sẽ được đáp ứng sau khi use case được thực hiện thành công.

10. Một số lưu ý khi trích các use case là gì?

* Tập trung vào mục tiêu của người dùng, không phải chi tiết kỹ thuật.
* Sử dụng ngôn ngữ rõ ràng, dễ hiểu.
* Xác định rõ các actor và tương tác của họ với hệ thống.
* Mô tả đầy đủ các bước thực hiện và các luồng sự kiện thay thế.
* Đảm bảo tính nhất quán và đầy đủ của các use case.

**Câu hỏi tình huống:**

1. Khách hàng không hiểu rõ các thuật ngữ kỹ thuật trong tài liệu yêu cầu. Là trưởng nhóm phát triển, bạn sẽ làm gì?

* Giải thích rõ ràng: Sử dụng ngôn ngữ đơn giản, dễ hiểu để giải thích các thuật ngữ kỹ thuật.
* Cung cấp ví dụ: Sử dụng các ví dụ thực tế để minh họa các khái niệm.
* Tạo bảng chú giải: Xây dựng bảng chú giải các thuật ngữ kỹ thuật và giải thích ý nghĩa của chúng.
* Tổ chức buổi thảo luận: Tổ chức buổi thảo luận để giải đáp thắc mắc của khách hàng.
* Sử dụng hình ảnh và sơ đồ: Sử dụng hình ảnh và sơ đồ để minh họa các khái niệm.

2. Trong quá trình lấy yêu cầu, khách hàng liên tục thay đổi ý kiến về chức năng cần thiết. Đội phát triển nên xử lý ra sao?

* Ghi nhận tất cả thay đổi: Ghi nhận lại tất cả các thay đổi yêu cầu của khách hàng.
* Đánh giá tác động: Đánh giá tác động của các thay đổi đến thời gian, chi phí và nguồn lực của dự án.
* Thảo luận với khách hàng: Thảo luận với khách hàng về lý do thay đổi và các chi phí phát sinh.
* Ưu tiên các thay đổi quan trọng: Ưu tiên thực hiện các thay đổi quan trọng nhất.
* Cập nhật tài liệu yêu cầu: Cập nhật tài liệu yêu cầu để phản ánh các thay đổi.

3. Một dự án phần mềm quản lý bán hàng gặp vấn đề khi các yêu cầu được mô tả quá chung chung, khó thực hiện. Đội phát triển cần làm gì để khắc phục?

* Phỏng vấn chi tiết: Phỏng vấn khách hàng để thu thập thông tin chi tiết về các yêu cầu.
* Xây dựng use case: Xây dựng use case để mô tả chi tiết các tương tác giữa người dùng và hệ thống.
* Xây dựng prototype: Xây dựng prototype để minh họa các chức năng của hệ thống.
* Xác định các ràng buộc: Xác định các ràng buộc về kỹ thuật, thời gian và chi phí.
* Tổ chức buổi thảo luận: Tổ chức buổi thảo luận để thống nhất các yêu cầu.

4. Khách hàng đưa ra yêu cầu không rõ ràng và chỉ có thể mô tả một cách sơ lược. Làm thế nào để thu thập yêu cầu đầy đủ từ khách hàng?

* Sử dụng kỹ thuật đặt câu hỏi: Sử dụng các kỹ thuật đặt câu hỏi như "5 Whys" để đào sâu vào vấn đề.
* Sử dụng kỹ thuật Brainstorming: Sử dụng kỹ thuật Brainstorming để thu thập ý tưởng từ khách hàng.
* Sử dụng kỹ thuật Use Case: Sử dụng kỹ thuật Use Case để mô tả chi tiết các tương tác giữa người dùng và hệ thống.
* Sử dụng kỹ thuật Prototype: Sử dụng kỹ thuật Prototype để minh họa các chức năng của hệ thống.
* Quan sát quy trình làm việc: Quan sát quy trình làm việc của khách hàng để hiểu rõ nhu cầu của họ.

5. Trong quá trình xây dựng mô hình nghiệp vụ, một số use case bị trùng lặp về chức năng. Bạn sẽ xử lý tình huống này như thế nào?

* Phân tích các use case: Phân tích các use case trùng lặp để xác định điểm chung và điểm khác biệt.
* Hợp nhất các use case: Hợp nhất các use case trùng lặp thành một use case duy nhất.
* Sử dụng quan hệ Include: Sử dụng quan hệ Include để chia sẻ chức năng giữa các use case.
* Tổ chức buổi thảo luận: Tổ chức buổi thảo luận để thống nhất các use case.
* Cập nhật mô hình nghiệp vụ: Cập nhật mô hình nghiệp vụ để phản ánh các thay đổi.

6. Đội phát triển và khách hàng có ý kiến khác nhau về ý nghĩa của một số từ khóa chuyên môn. Là trưởng dự án, bạn sẽ làm gì?

* Tổ chức buổi thảo luận: Tổ chức buổi thảo luận để các bên trình bày ý kiến.
* Tham khảo tài liệu chuyên ngành: Tham khảo tài liệu chuyên ngành để tìm định nghĩa chính xác.
* Đưa ra định nghĩa thống nhất: Đưa ra định nghĩa thống nhất cho các từ khóa chuyên môn.
* Ghi nhận lại các định nghĩa: Ghi nhận lại các định nghĩa để sử dụng trong suốt dự án.
* Cập nhật bảng chú giải: Cập nhật bảng chú giải các thuật ngữ chuyên môn.

7. Sau khi hoàn thành việc trích các use case, đội phát triển nhận ra rằng một số chức năng quan trọng bị bỏ sót. Hãy đề xuất cách giải quyết.

* Xem xét lại tài liệu yêu cầu: Xem xét lại tài liệu yêu cầu để xác định các chức năng bị bỏ sót.
* Phỏng vấn khách hàng: Phỏng vấn khách hàng để thu thập thông tin chi tiết về các chức năng bị bỏ sót.
* Bổ sung use case: Bổ sung các use case để mô tả các chức năng bị bỏ sót.
* Đánh giá tác động: Đánh giá tác động của việc bổ sung use case đến thời gian, chi phí và nguồn lực của dự án.
* Thảo luận với khách hàng: Thảo luận với khách hàng về việc bổ sung use case.

8. Trong quá trình xây dựng mô hình nghiệp vụ, khách hàng yêu cầu bổ sung thêm một số chức năng mới. Đội phát triển nên làm gì?

* Ghi nhận yêu cầu: Ghi nhận lại yêu cầu bổ sung chức năng của khách hàng.
* Đánh giá tác động: Đánh giá tác động của việc bổ sung chức năng đến thời gian, chi phí và nguồn lực của dự án.
* Thảo luận với khách hàng: Thảo luận với khách hàng về khả năng thực hiện và các chi phí phát sinh.
* Cập nhật mô hình nghiệp vụ: Cập nhật mô hình nghiệp vụ để phản ánh các thay đổi.
* Lập kế hoạch bổ sung: Lập kế hoạch chi tiết cho việc bổ sung chức năng.

9. Một nhóm phát triển gặp khó khăn trong việc mô tả chi tiết các use case vì chưa hiểu rõ quy trình nghiệp vụ. Hãy đề xuất giải pháp.

* Tìm hiểu quy trình nghiệp vụ: Tìm hiểu kỹ lưỡng quy trình nghiệp vụ của khách hàng.
* Quan sát quy trình làm việc: Quan sát quy trình làm việc của khách hàng để hiểu rõ các bước thực hiện.
* Phỏng vấn chuyên gia: Phỏng vấn các chuyên gia về quy trình nghiệp vụ.
* Sử dụng tài liệu tham khảo: Sử dụng tài liệu tham khảo về quy trình nghiệp vụ.
* Xây dựng sơ đồ quy trình: Xây dựng sơ đồ quy trình để mô tả chi tiết các bước thực hiện.

10. Khách hàng yêu cầu thêm một chức năng mới khi hệ thống đã bước vào giai đoạn thiết kế chi tiết. Đội phát triển cần xử lý như thế nào?

* Đánh giá tác động: Đánh giá tác động của việc bổ sung chức năng đến thiết kế hệ thống.
* Thảo luận với khách hàng: Thảo luận với khách hàng về khả năng thực hiện và các chi phí phát sinh.
* Cập nhật thiết kế: Cập nhật thiết kế hệ thống để phản ánh các thay đổi.
* Lập kế hoạch thay đổi: Lập kế hoạch chi tiết cho việc thực hiện thay đổi.
* Kiểm tra kỹ lưỡng: Kiểm tra kỹ lưỡng sau khi thực hiện thay đổi để đảm bảo không ảnh hưởng đến các chức năng khác.