

BÀI TẬP BUỔI 1

I. Trắc nghiệm

Câu hỏi 1: D. Cả A, B và C (Phần mềm hệ thống, phần mềm ứng dụng, phần mềm nhúng)

Câu hỏi 2: C. Ứng dụng các phương pháp khoa học để phát triển phần mềm

Câu hỏi 3: C. 5 (Quy trình gồm: Lấy yêu cầu, Thiết kế, Lập trình, Kiểm thử, Bảo trì)

Câu hỏi 4: C. Cập nhật phần mềm để phù hợp với thay đổi môi trường

Câu hỏi 5: C. 60% (Chi phí bảo trì chiếm phần lớn trong vòng đời phần mềm)

Câu hỏi 6: D. Cả A và C (Thiếu nhân lực và thay đổi công nghệ)

Câu hỏi 7: D. Chức năng đăng nhập (Vì đây là yêu cầu chức năng, không phải phi chức năng)

Câu hỏi 8: D. Khi được khách hàng chấp nhận và đưa vào sử dụng

Câu hỏi 9: D. Tất cả đều đúng (Vượt chi phí, trễ thời hạn, lỗi sau bàn giao, thiếu công cụ, không có đội kiểm thử)

Câu hỏi 10: B. 3 (Phần mềm hệ thống, phần mềm ứng dụng, phần mềm nhúng)

II. Câu hỏi ngắn

1. Phần mềm là gì?

Phần mềm là tập hợp các hướng dẫn, chương trình được viết để máy tính thực thi nhằm thực hiện các chức năng hoặc nhiệm vụ cụ thể

2. Công nghệ phần mềm là gì?

Công nghệ phần mềm là lĩnh vực nghiên cứu, phát triển và áp dụng các phương pháp có hệ thống để xây dựng phần mềm chất lượng cao

3. Các loại phần mềm chính là gì?

Phần mềm được chia thành ba loại chính:

- Phần mềm hệ thống: Hệ điều hành, trình điều khiển thiết bị.
 - Phần mềm ứng dụng: Các chương trình phục vụ công việc hoặc nhu cầu của người dùng như Microsoft Office, trình duyệt web.
 - Phần mềm nhúng: Phần mềm điều khiển các thiết bị phần cứng như máy giặt, điều hòa
4. Tại sao công nghệ phần mềm lại quan trọng?

Công nghệ phần mềm giúp tạo ra phần mềm đúng yêu cầu khách hàng, đảm bảo phần mềm có thể bảo trì và nâng cấp dễ dàng, đồng thời giúp giảm thiểu thời gian và chi phí phát triển

5. Quy trình phát triển phần mềm gồm những giai đoạn nào?

Gồm 5 giai đoạn chính:

- Lấy yêu cầu: Thu thập và phân tích yêu cầu từ khách hàng.
 - Thiết kế: Lên kế hoạch và cấu trúc hệ thống phần mềm.
 - Lập trình: Chuyển đổi thiết kế thành mã nguồn thực thi.
 - Kiểm thử: Đảm bảo phần mềm hoạt động đúng chức năng.
 - Bảo trì: Khắc phục lỗi và nâng cấp phần mềm sau khi triển khai
6. Khía cạnh kinh tế của công nghệ phần mềm là gì?

Công nghệ phần mềm giúp tối ưu hóa chi phí vận hành doanh nghiệp, tăng năng suất lao động và kiểm soát chi phí tốt hơn trong các dự án phần mềm lớn

7. Khía cạnh công nghệ của công nghệ phần mềm là gì?

Phần mềm cần được cải tiến liên tục để đáp ứng sự phát triển nhanh chóng của công nghệ, giúp kết nối các hệ thống phức tạp và xử lý khối lượng dữ liệu lớn

8. Khía cạnh bảo trì của công nghệ phần mềm là gì?

Sau khi phần mềm triển khai, bảo trì là hoạt động quan trọng để đảm bảo phần mềm hoạt động ổn định, cập nhật theo nhu cầu người dùng. Bảo trì phần mềm chiếm khoảng 60% tổng chi phí vòng đời phần mềm

9. Các nguyên nhân chính gây trễ thời hạn khi phát triển phần mềm là gì?

Các nguyên nhân phổ biến gồm:

- Không ước lượng chính xác yêu cầu và khối lượng công việc.
 - Thay đổi yêu cầu trong quá trình phát triển.
 - Thiếu nhân lực hoặc sự phối hợp kém giữa các thành viên trong nhóm
10. Bảo trì phần mềm bao gồm những hoạt động nào?

Hoạt động bảo trì bao gồm:

- Sửa lỗi: Khắc phục các lỗi phát sinh sau khi triển khai.
- Cập nhật phần mềm: Điều chỉnh để phù hợp với thay đổi của môi trường hệ thống.
- Nâng cấp tính năng: Thêm hoặc cải tiến các chức năng để đáp ứng nhu cầu người dùng

III. Câu hỏi thảo luận nhóm

1. Phân biệt phần mềm hệ thống và phần mềm ứng dụng

- Phần mềm hệ thống: Là phần mềm dùng để quản lý và điều khiển phần cứng của máy tính. Ví dụ: Hệ điều hành (Windows, Linux), trình điều khiển thiết bị.
- Phần mềm ứng dụng: Là phần mềm phục vụ các nhu cầu cụ thể của người dùng. Ví dụ: Microsoft Office, trình duyệt web, phần mềm kế toán

2. Thảo luận về vai trò của công nghệ phần mềm trong lĩnh vực tài chính

- Hỗ trợ quản lý giao dịch tài chính chính xác và nhanh chóng.
 - Đảm bảo bảo mật dữ liệu tài chính của khách hàng.
 - Hỗ trợ phân tích dữ liệu lớn để đưa ra quyết định đầu tư hiệu quả.
 - Ứng dụng AI để phát hiện gian lận tài chính và tối ưu hóa quy trình
- ### 3. Nêu các thách thức thường gặp trong bảo trì phần mềm
- Chi phí cao: Bảo trì chiếm phần lớn chi phí vòng đời phần mềm (~60%).
 - Tích hợp với hệ thống cũ: Nhiều phần mềm phải hoạt động cùng với hệ thống đã lỗi thời.
 - Thay đổi yêu cầu: Người dùng thường yêu cầu cập nhật và bổ sung tính năng mới.
 - Bảo mật: Các lỗ hổng bảo mật phải được vá kịp thời để tránh rủi ro
- ### 4. Vì sao phần mềm thương mại điện tử cần được bảo trì thường xuyên?
- Cập nhật để thích nghi với xu hướng mua sắm mới.
 - Đảm bảo hệ thống bảo mật trước các cuộc tấn công mạng.
 - Tối ưu hiệu suất để đáp ứng lượng lớn người dùng.
 - Khắc phục lỗi và cải tiến trải nghiệm khách hàng
- ### 5. Phân tích những vấn đề khi yêu cầu khách hàng liên tục thay đổi trong quá trình phát triển phần mềm
- Làm chậm tiến độ: Các thay đổi làm gián đoạn quy trình phát triển.
 - Tăng chi phí: Phải điều chỉnh lại thiết kế, mã nguồn, kiểm thử.
 - Gây nhầm lẫn: Đội phát triển khó kiểm soát được phạm vi dự án (Scope Creep).
 - Ảnh hưởng chất lượng: Phần mềm có thể thiếu ổn định do thay đổi liên tục
- ### 6. So sánh chi phí phát triển và chi phí bảo trì phần mềm
- Chi phí phát triển: Gồm chi phí thiết kế, lập trình, kiểm thử, triển khai ban đầu.
 - Chi phí bảo trì: Gồm sửa lỗi, cập nhật, tối ưu hiệu suất, tích hợp với hệ thống khác.
 - Bảo trì tốn kém hơn: Khoảng 60% tổng chi phí vòng đời phần mềm dành cho bảo trì
- ### 7. Phân biệt các loại yêu cầu trong phát triển phần mềm (chức năng và phi chức năng)
- Yêu cầu chức năng: Xác định phần mềm phải làm gì (VD: Đăng nhập, xử lý đơn hàng).
 - Yêu cầu phi chức năng: Xác định cách phần mềm thực hiện (VD: Hiệu suất, bảo mật, khả năng mở rộng)
- ### 8. Thảo luận về các mô hình quy trình phát triển phần mềm phổ biến
- Waterfall: Quy trình tuần tự, dễ kiểm soát nhưng khó thay đổi yêu cầu.
 - Agile: Linh hoạt, phù hợp với yêu cầu thay đổi liên tục.
 - Scrum: Áp dụng phương pháp Agile với các giai đoạn phát triển ngắn.
 - Spiral Model: Kết hợp giữa phát triển lặp và quản lý rủi ro
- ### 9. Đề xuất giải pháp giảm thiểu lỗi phần mềm sau khi bàn giao
- Kiểm thử kỹ lưỡng trước khi phát hành.

- Tự động hóa kiểm thử để phát hiện lỗi nhanh hơn.
- Thu thập phản hồi từ người dùng và cập nhật kịp thời.
- Áp dụng DevOps để liên tục giám sát và cải tiến phần mềm

10. Vai trò của đội kiểm thử trong quy trình phát triển phần mềm

- Đảm bảo phần mềm đáp ứng yêu cầu và không có lỗi nghiêm trọng.
- Phát hiện sớm các vấn đề về hiệu suất, bảo mật.
- Giúp giảm chi phí bảo trì bằng cách sửa lỗi ngay từ đầu.
- Hỗ trợ kiểm thử tự động để tăng tốc độ phát triển

IV. Câu hỏi tình huống

Tình huống 1: Phần mềm quản lý tài chính phát sinh lỗi khi xử lý giao dịch lớn

- Kiểm tra lại thuật toán xử lý giao dịch, đặc biệt là giới hạn số học.
- Tối ưu hiệu suất xử lý dữ liệu, sử dụng kiểu dữ liệu phù hợp.
- Bổ sung kiểm thử với giao dịch có giá trị lớn để đảm bảo độ chính xác.
- Phát hành bản vá nhanh chóng để khắc phục lỗi

Tình huống 2: Bổ sung tính năng quản lý kho thuốc khi đã vào giai đoạn kiểm thử

- Đánh giá mức độ ảnh hưởng đến hệ thống và tiến độ dự án.
- Nếu ảnh hưởng lớn, đề xuất triển khai tính năng trong bản cập nhật sau.
- Nếu có thể tích hợp ngay, thực hiện kiểm thử hồi quy (Regression Testing) để đảm bảo ổn định

Tình huống 3: Nhóm phát triển trễ tiến độ do không hiểu rõ yêu cầu khách hàng

- Tổ chức lại các buổi họp với khách hàng để làm rõ yêu cầu.
- Tăng cường tài liệu đặc tả yêu cầu và hướng dẫn chi tiết.
- Phân công công việc rõ ràng, giám sát tiến độ thường xuyên

Tình huống 4: Phần mềm quản lý thư viện có giao diện khó sử dụng

- Thu thập phản hồi cụ thể từ người dùng.
- Điều chỉnh giao diện theo nguyên tắc UI/UX để cải thiện trải nghiệm.
- Triển khai thử nghiệm với nhóm người dùng trước khi cập nhật chính thức

Tình huống 5: Dự án vượt ngân sách do kéo dài thời gian

- Xác định nguyên nhân cụ thể (yêu cầu thay đổi, hiệu suất làm việc kém...).
- Áp dụng mô hình quản lý Agile để theo dõi tiến độ tốt hơn.
- Tăng cường kiểm soát phạm vi dự án (Scope Management)

Tình huống 6: Lỗi nhỏ nhưng chi phí sửa cao trong phần mềm khách sạn

- Đánh giá mức độ ảnh hưởng: Nếu lỗi không gây gián đoạn, có thể hoãn sửa.
- Nếu ảnh hưởng đến bảo mật hoặc trải nghiệm người dùng, ưu tiên khắc phục.
- Xem xét giải pháp tạm thời nếu sửa lỗi tốn kém

Tình huống 7: Khách hàng yêu cầu hoàn thành sớm hơn 1 tháng

- Đánh giá khả năng tăng nhân sự hoặc làm việc ngoài giờ.
- Nếu không khả thi, thỏa thuận lại thời gian hoặc phạm vi tính năng.
- Sử dụng nguyên tắc ưu tiên để hoàn thành phần quan trọng trước

Tình huống 8: Công ty nhỏ liên tục thay đổi công nghệ, khiến dự án kéo dài

- Ổn định công nghệ ngay lập tức, không thay đổi thêm.
- Đào tạo đội ngũ để tối ưu hóa công nghệ đã chọn.
- Áp dụng nguyên tắc tối thiểu hóa tính năng (MVP) để hoàn thành sản phẩm sớm

Tình huống 9: Phát hiện lỗi bảo mật nghiêm trọng sau khi bàn giao

- Ngay lập tức phát hành bản vá bảo mật.
- Thông báo khách hàng về nguy cơ và hướng dẫn biện pháp phòng tránh.
- Tăng cường kiểm thử bảo mật để tránh lỗi tương tự trong tương lai

Tình huống 10: Khách hàng thay đổi quy trình sản xuất sau khi triển khai phần mềm

- Phân tích tác động của thay đổi lên phần mềm.
- Nếu thay đổi lớn, đề xuất phát triển bản cập nhật riêng.
- Đảm bảo hệ thống vận hành song song để không ảnh hưởng đến hoạt động sản xuất