|  |  |
| --- | --- |
| **Nhóm** | **Thành viên** |
| 10 | 1. Phan Minh Huy  2. Võ Quốc Anh  3. Phạm Lê Tiến Dũng  4.Nguyễn Minh Chiến |

1. **Vòng đời của phần mềm gồm những giai đoạn?**

Bao gồm các giai đoạn:

+ Thu thập yêu cầu

+ Phân tích yêu cầu

+ Thiết kế phần mềm

+ Phát triển phần mềm

+ Kiểm thử và triển khai

+ Bảo trì và nâng cấp

**Chi phí của giai đoạn nào nhiều hơn? Tạo sao?**

+ Chi phí của giai đoạn phát triển phần mềm thường nhiều hơn các giai đoạn khác. Lý do là trong giai đoạn này, cần có sự đầu tư lớn vào việc thuê nhân lực, mua sắm công cụ và phần cứng cần thiết, đảm bảo môi trường phát triển phần mềm và kiểm thử. Ngoài ra, giai đoạn phát triển còn kéo dài trong khoảng thời gian dài hơn so với các giai đoạn khác do yêu cầu công việc lập trình, kiểm thử và tích hợp phần mềm.

**Giai đoạn phát triển phần mềm gồm có những công đoạn nào?**

+ Phân tich yêu cầu  
+ Đặc tả  
+ Thiết kế   
+ Lập trình   
+ Kiểm thử  
+ Vận hành và bảo trì

**Bạn đang trả lời các công đoạn này dựa vào mô hình phát triển phần mềm nào?**

+ Các công đoạn trên dựa vào mô hình phát triển phần mềm waterfall

1. **Hãy nêu một số mô hình phát triển phần mềm mà bạn biết?**

+ Mô hình Water fall

+ Mô hình Spiral

+ Mô hình Agile

+ Mô hình V-Model

+ Mô hình DevOps

+ Mô hình Incremental

**Mô hình phát triển phần mềm nào được sử dụng khá phổ biến hiện nay? Tại sao mô hình này được ưu tiên sử dụng?**

+ Mô hình phát triển phần mềm được sử dụng phổ biến hiện nay là mô hình Agile

Lí do mô hình này được ưu tiên sử dụng:

+ Sự linh hoạt

+ Tập trung vào giá trị khách hàng

+ Phân chia dự án thành các đợt nhỏ

+ Kiểm soát rủi ro và đánh giá liên tục

+ Sự tương tác và cộng tác

+ Đội ngũ tự quản lý

**Có những giai đoạn kiểm thử nào trong mô hình chữ V? Những giai đoạn này được thực hiện tương ứng thời điểm nào trong quá trình phát triển phần mềm?**

Gồm các giai đoạn:

+ Kiểm tra yêu cầu (Requirements Testing)

+ Kiểm thử hệ thống (System Testing)

+ Kiểm thử tích hợp (Integration Testing)

+ Kiểm thử đơn vị (Unit Testing)

+ Kiểm thử hệ thống (System Testing)

1. **Lỗi phần mềm là gì?**

+ Lỗi phần mềm là một vấn đề hoặc sự cố xảy ra trong quá trình phát triển hoặc sử dụng phần mềm, dẫn đến hoạt động không đúng, không mong muốn hoặc không ổn định của phần mềm đó. Các lỗi phần mềm có thể xuất hiện trong mọi giai đoạn của quá trình phát triển phần mềm, từ thiết kế, lập trình, kiểm thử cho đến triển khai và sử dụng.

**Lỗi phần mềm xảy ra ở giai đoạn nào (Thu thập yêu cầu (đặc tả), Thiết kế, hay Lập trình (coding)) ? Cho ví dụ về lỗi phần mềm, cho biết lỗi này xảy ra ở giai đoạn nào**

+ Lỗi phần mềm có thể xảy ra ở bất kỳ giai đoạn nào trong quy trình phát triển phần mềm, từ thu thập yêu cầu (đặc tả), thiết kế đến lập trình (coding). Dưới đây là một số nguyên nhân phổ biến gây ra lỗi ở mỗi giai đoạn:

+ Thu thập yêu cầu (đặc tả): Lỗi có thể xuất hiện nếu yêu cầu không được hiểu rõ hoặc ghi chép không đầy đủ. Sự không rõ ràng hoặc thiếu sót trong việc thu thập yêu cầu có thể dẫn đến lỗi trong quá trình phát triển phần mềm.

+ Thiết kế: Lỗi có thể phát sinh nếu thiết kế không đáp ứng đúng yêu cầu hoặc không được thực hiện một cách chính xác. Thiết kế không tốt có thể dẫn đến các vấn đề về hiệu suất, bảo mật, tính khả quan sát và khả năng mở rộng của phần mềm.

+ Lập trình (coding): Lỗi trong mã nguồn phần mềm là phổ biến nhất. Các lỗi cú pháp, lỗi logic, lỗi kiểm soát luồng chương trình hoặc lỗi xử lý dữ liệu có thể xảy ra trong quá trình viết mã. Các lỗi này thường bao gồm việc không xử lý đúng các trường hợp đặc biệt, kiểm tra không chính xác hoặc thiếu kiểm tra, và nhiều lỗi khác.

- Một số lỗi khác:

+ Thử nghiệm sai hoặc không đầy đủ

+ Các lỗi về tài liệu (viết không đầy đủ hoặc không chính xác)

+ Do thay đổi môi trường, công nghệ

+ Do thay đổi phần cứng, hạ tầng phức tạp

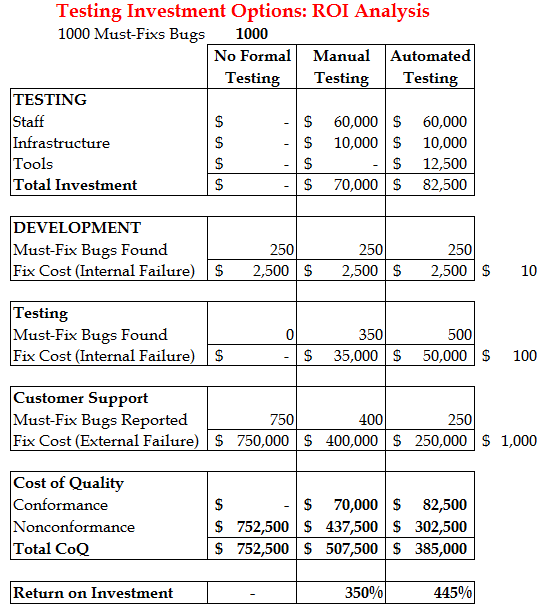
+ Tương tác nhiều hệ thống

1. **Định nghĩa kiểm thử phần mềm?**

+ Kiểm thử phần mềm là quá trình đánh giá và xác nhận tính chất chất lượng và hiệu suất của một phần mềm hoặc hệ thống phần mềm. Nó bao gồm việc kiểm tra và xác minh các tính năng, chức năng và yêu cầu của phần mềm, đảm bảo rằng nó hoạt động như được thiết kế và đáp ứng được các tiêu chuẩn chất lượng.

+ Quá trình kiểm thử phần mềm có thể bao gồm việc tạo các ca kiểm thử, thiết lập môi trường kiểm thử, thực hiện các bước kiểm tra và ghi nhận kết quả. Nó có thể được thực hiện tự động hoặc thủ công, bao gồm các phương pháp như kiểm thử đơn vị, kiểm thử hệ thống, kiểm thử tích hợp, kiểm thử chấp nhận và nhiều hơn nữa.

1. **Giải thích tại sao kiểm thử là cần thiết thông qua bảng sau:**



Phân tích ROI (Return on Investment) là một phương pháp được sử dụng để đánh giá lợi tức tài chính hoặc khả năng sinh lời của một khoản đầu tư. Nó giúp các nhà đầu tư đánh giá các khoản lãi hoặc lỗ tiềm năng liên quan đến các lựa chọn đầu tư khác nhau. Đây là một khuôn khổ chung để tiến hành phân tích ROI cho các lựa chọn đầu tư:

Xác định các lựa chọn đầu tư: Bắt đầu bằng cách xác định các lựa chọn đầu tư mà bạn muốn đánh giá. Chúng có thể bao gồm cổ phiếu, trái phiếu, bất động sản, quỹ tương hỗ hoặc bất kỳ phương tiện đầu tư nào khác.

Thu thập dữ liệu: Thu thập dữ liệu có liên quan cho từng tùy chọn đầu tư, bao gồm hiệu suất lịch sử, lợi nhuận dự kiến, chi phí liên quan (chẳng hạn như phí hoặc hoa hồng) và bất kỳ yếu tố nào khác có thể ảnh hưởng đến lợi tức của khoản đầu tư.

Tính toán ROI: Tính toán ROI cho từng tùy chọn đầu tư. Công thức cơ bản cho ROI là:

ROI = (Lợi nhuận ròng / Chi phí đầu tư) x 100

Lợi nhuận ròng = Tổng lợi nhuận - Chi phí đầu tư

Lợi nhuận ròng thể hiện các khoản lãi hoặc lỗ do khoản đầu tư tạo ra, trong khi chi phí đầu tư bao gồm số tiền đầu tư ban đầu.

Xem xét Khung thời gian: Xác định khung thời gian cho phân tích đầu tư của bạn. Một số khoản đầu tư có thể mang lại lợi nhuận ngắn hạn, trong khi những khoản đầu tư khác có thể phù hợp hơn cho các chiến lược dài hạn. Xem xét lợi nhuận kỳ vọng và khung thời gian liên quan đến từng lựa chọn đầu tư.

Đánh giá rủi ro: Đánh giá rủi ro liên quan đến từng lựa chọn đầu tư. Các khoản đầu tư có lợi nhuận tiềm năng cao hơn thường đi kèm với rủi ro cao hơn. Xem xét các yếu tố như biến động thị trường, điều kiện kinh tế và rủi ro cụ thể liên quan đến khoản đầu tư.

So sánh các lựa chọn đầu tư: So sánh ROI được tính toán của các lựa chọn đầu tư khác nhau. Hãy tính đến mức độ rủi ro, thời hạn và mục tiêu đầu tư của bạn. Xem xét các yếu tố định tính khác, chẳng hạn như lợi ích đa dạng hóa hoặc sự liên kết của tùy chọn đầu tư với kế hoạch tài chính tổng thể của bạn.

Đưa ra quyết định sáng suốt: Dựa trên phân tích ROI và mức độ chấp nhận rủi ro của bạn, hãy đưa ra quyết định đầu tư sáng suốt. Chọn các tùy chọn đầu tư phù hợp với mục tiêu tài chính, khẩu vị rủi ro và thời hạn đầu tư của bạn.

1. **Phân biệt: Xác minh – Verification “Are we building the product right?” và Thẩm định – Validation “Are we building the right product?”**

Xác minh (Verification): "Are we building the product right?" là quá trình kiểm tra xem phần mềm đã được xây dựng đúng cách không. Xác minh tập trung vào việc xác định liệu phần mềm có tuân thủ đúng các yêu cầu đã đặt ra hay không. Nó nhằm đảm bảo rằng phần mềm đã được xây dựng đúng theo các thông số kỹ thuật và yêu cầu đã được đặt ra từ trước.

Thẩm định (Validation): "Are we building the right product?" là quá trình kiểm tra xem chúng ta có đang xây dựng sản phẩm đúng không. Thẩm định tập trung vào việc đánh giá xem phần mềm đã đáp ứng được nhu cầu thực tế và mong đợi của người sử dụng hay không. Nó nhằm đảm bảo rằng phần mềm đang được xây dựng có tính hợp lý và đáp ứng được mục tiêu kinh doanh hoặc yêu cầu của người dùng.

**QA, QC là gì? Vai trò của QA, QC? Ai thực hiện Verification? Ai thực hiện Validation?**

+ Quality Assurance (QA): QA là quá trình và hoạt động được thực hiện để đảm bảo rằng quy trình sản xuất, phát triển hoặc cung cấp một sản phẩm hoặc dịch vụ đáp ứng các tiêu chuẩn chất lượng đã được đặt ra. Vai trò chính của QA là tạo ra các quy trình, quy chuẩn và hệ thống để đảm bảo chất lượng trong toàn bộ quá trình.

+ Quality Control (QC): QC là quá trình kiểm soát chất lượng được thực hiện để đảm bảo rằng sản phẩm hoặc dịch vụ đáp ứng các tiêu chuẩn chất lượng đã được đề ra. Vai trò chính của QC là kiểm tra, đo lường và kiểm soát các yếu tố liên quan đến chất lượng sản phẩm hoặc dịch vụ trong quá trình sản xuất hoặc cung cấp.

Verification (Xác thực): Verification là quá trình kiểm tra xem sản phẩm, hệ thống hoặc quy trình có tuân thủ các yêu cầu được đặt ra ban đầu hay không. Nó đảm bảo rằng mọi yêu cầu đã được triển khai đúng cách và theo đúng thông số kỹ thuật.

Validation (Xác nhận): Validation là quá trình kiểm tra xem sản phẩm hoặc dịch vụ có đáp ứng yêu cầu và mong đợi của người dùng cuối hay không. Nó kiểm tra tính chính xác, hiệu suất và đáng tin cậy của sản phẩm hoặc dịch vụ.

**Tester là QA hay QC? Tester thực hiện Verification hay Validation?**

+ Trong ngữ cảnh phần mềm, Tester thường được coi là một phần của quá trình kiểm thử và thường thuộc vai trò của QA (Quality Assurance). Tester thực hiện các hoạt động kiểm thử để tìm lỗi, đánh giá tính năng và đảm bảo chất lượng của sản phẩm phần mềm.

+ Về verification và validation, Tester thường thực hiện verification, tức là kiểm tra xem sản phẩm phần mềm tuân thủ các yêu cầu đã được đặt ra. Tester xác minh rằng sản phẩm được phát triển đúng theo các thông số kỹ thuật, tiêu chuẩn và quy trình đã được xác định.

+ Tuy nhiên, việc thực hiện validation, tức là kiểm tra xem sản phẩm phần mềm có đáp ứng yêu cầu và mong đợi của người dùng cuối hay không, thường thuộc trách nhiệm của người dùng cuối hoặc người sử dụng cuối cùng của sản phẩm. Validation thường liên quan đến việc đánh giá sự phù hợp và hiệu quả của sản phẩm trong ngữ cảnh thực tế sử dụng.

+ Tóm lại, Tester thường thuộc vai trò QA và thực hiện verification, trong khi validation thường do người dùng cuối hoặc người sử dụng cuối cùng thực hiện

1. **Thái độ của Tester?**

Thái độ của một tester trong lĩnh vực kiểm thử phần mềm có thể ảnh hưởng đáng kể đến quá trình và kết quả của công việc. Dưới đây là một số đặc điểm và thái độ quan trọng mà người kiểm thử nên có:

1. Tinh thần kiểm thử: Tester cần có tinh thần tỉ mỉ, kiên nhẫn và cẩn thận. Họ phải quan tâm đến chi tiết nhỏ và có khả năng tìm ra các lỗi tiềm ẩn trong phần mềm.

2. Tư duy phản biện: Tester nên có tư duy phản biện, luôn đặt câu hỏi và không chấp nhận mặc kệ những điều không rõ ràng. Họ phải có khả năng phân tích và đưa ra luận điểm logic trong quá trình kiểm thử.

3. Sáng tạo: Tester nên có khả năng tư duy sáng tạo để tạo ra các kịch bản kiểm thử đa dạng và phát hiện các trường hợp kiểm thử tiềm năng mà người khác có thể bỏ qua.

4. Tinh thần đồng đội: Tester thường là một phần của một nhóm phát triển phần mềm. Họ cần có khả năng làm việc trong nhóm, chia sẻ thông tin và ý kiến với các thành viên khác của nhóm để đảm bảo chất lượng tổng thể của sản phẩm.

5. Kiên nhẫn: Tester thường phải xử lý với các vấn đề khó khăn và phải kiểm tra lại nhiều lần để đảm bảo tính toàn vẹn của phần mềm. Kiên nhẫn là một yếu tố quan trọng để không bỏ qua các lỗi quan trọng.

6. Khả năng giao tiếp: Tester cần có khả năng giao tiếp tốt để truyền đạt thông tin kiểm thử một cách rõ ràng và hiệu quả. Họ cần thể hiện khả năng báo cáo kết quả kiểm thử và tương tác với các thành viên khác trong nhóm.

7. Tư duy kiểm thử: Tester nên có khả năng tư duy kiểm thử, đặt mình vào tư duy của người dùng cuối và tìm hiểu cách sử dụng phần mềm từ góc nhìn của họ. Điều này giúp họ tìm ra các lỗi và vấn đề có thể phát sinh trong quá trình sử dụng thực tế.

Tóm lại, thái độ của tester cần làm việc tỉ mỉ, phản biện, sáng tạo và có tinh thần đồng đội. Đồng thời, họ cũng cần kiên nhẫn, có khả năng giao tiếp tốt và tư duy kiểm thử để đảm bảo chất lượng phần mềm được kiểm thử.