* **Hướng dẫn kịch bản Test Case tính khả dụng (Usability Testing)**
* Kiểm tra tính thân thiện với người dùng
* Kiểm tra tính dễ sử dụng với người dùng
* Nhằm thiết lập tính hiệu quả và dễ sử dụng cho sản phẩm thông qua kiểm tra
* Kịch bản

1. Nội dung trang web chính xác, không lỗi chính tả và ngữ pháp
2. Tất cả các Font chữ theo đúng yêu cầu
3. Tất cả các văn bản, Text được căn chỉnh chính xác
4. Tất cả các thông báo báo lỗi phải chính xác (không phải chính tả và ngữ pháp). Thông báo lỗi phải phù hợp với từng trường, từng nhãn, …
5. Tất cả các trường phải được căn chỉnh chính xác
6. Tất cả các nút (Button) phải đúng định dạng và kích thước
7. Các trang khác đều liên kết với trang chủ (Home Page)
8. Các trường không khả dụng nên để màu xám (grayout)
9. Cần hiển thị thông điệp thông báo khi có hoạt động update, delete
10. Kiểm tra hiển thị ở các độ phân giải khác nhau
11. Kiểm tra các Tab có hoạt động đúng không
12. Kiểm tra các thanh cuộn (scroll bar) hiển thị theo đúng yêu cầu không?
13. Nếu có thông báo lỗi được gửi đi thì cần có cả thông tin của người gửi
14. Mỗi trang web cần hiển thị tiêu đề (title page)
15. Tất cả các hộp văn bản, thả xuống, nút (Texbox, dropdown, button) có thể sử dụng phím tắt để thao tác (người dùng sẽ thực hiện trên bàn phím)
16. Kiểm tra trường dữ liệu (dropdown) có bị cắt bớt do kết thúc trường không, có đóng cứng không, có được quản lý bởi admin không,…

* **Hướng dẫn kịch bản Test Case kiểm tra chức năng (Functional Testing)**
* Kiểm tra các tính năng và hành vi hoạt động của sản phẩm để đảm bảo đúng với thông số kỹ thuật mô tả
* Bỏ qua cấu trúc bên trong của hệ thống mà chỉ tập trung vào giá trị vào / ra của hệ thống
* Xác minh xem sản phẩm có đáp ứng được các thông số chức năng đã được mô tả trong tài liệu phát triển hay không
* Kịch bản

1. Kiểm tra tất cả các trường bắt buộc => phải hiển thị dấu (\*)
2. Kiểm tra các trường không bắt buộc => không hiển thị báo lỗi
3. Các trường thời gian, ngày tháng năm tính toán đúng không?
4. Các trường không chấp nhận ký tự a,b,b, … => hiển thị báo lỗi không thích hợp
5. Kiểm tra số âm nếu là trường số
6. Kiểm tra trường hợp chia cho “0” => xử lý tính toán có đúng không?
7. Kiểm tra chiều dài tối đa của mỗi trường để đảm bảo dữ liệu không bị cắt ngắn
8. Khi thực hiện update, delete => cần thông báo xác nhận
9. Các trường tiền tệ hiển thị đúng định dạng
10. Các trường input => kiểm tra ký tự đặc biệt
11. Kiểm tra chức năng khi Time Out
12. Kiểm tra chức năng sắp xếp
13. Kiểm tra tài liệu tải lên => cần mở được và đúng
14. Kiểm tra người dùng có thể tải xuống các file đã tải lên
15. Kiểm tra chức năng email của hệ thống
16. Kiểm tra tập lệnh (java) sẽ hoạt động ở tất cả các trình duyệt khác nhau: IE, Firefox, chrome, safari, Opera, …)
17. Kiểm tra xem điều gì xảy ra khi người dùng xóa cookie, …?

* **Hướng dẫn kịch bản Test Case kiểm tra khả năng tương thích (Compatibility Testing)**
* Xác định xem phần mềm có tương thích với các yếu tố khác của hệ thống hay không. Ví dụ: trình duyệt, hệ điều hành, phần cứng, …
* Đánh giá xem phần mềm có hoạt động tốt hay không trong 1 trình duyệt, hệ điều hành, phần cứng hoặc phần mềm cụ thể
* Kịch bản

1. Đảm bảo wesite hoạt động đúng trên các trình duyệt khác nhau
2. Hình ảnh hiển thị trong các trình duyệt khác nhau: đúng / sai như thế nào?
3. Các font chữ có thể sử dụng trong các trình duyệt khác nhau không?
4. Ảnh động có hoạt động trên các trình duyệt khác nhau không?

* **Hướng dẫn kịch bản Test Case kiểm tra dữ liệu (DataBase Testing)**
* Quy trình thực hiện Database Testing
  + Thiết lập môi trường
  + Chạy thử nghiệm
  + Kiểm tra kết quả kiểm tra
  + Xác nhận theo kết quả mong đợi
  + Báo cáo kết quả cho các bên liên quan
* Các giai đoạn chính trong kiểm tra cơ sở dữ liệu là
  + Kiểm tra trạng thái ban đầu
  + Chạy thử nghiệm
  + Xác nhận kết quả theo kết quả mong đợi
  + Tạo kết quả
* Tạo test data tốt nhất
  + Thiết lập những dữ liệu không hợp lệ
  + Thiết lập các dữ liệu điều kiện biên
  + Tạo bộ dữ liệu đúng
  + Tạo bộ dữ liệu sai
  + Dữ liệu trống hoặc dữ liệu mặc định
  + Kiểm tra các dữ liệu hỏng/lỗi
  + Tạo ra các dữ liệu đầu vào có giá trị và sao chép nhân đôi
* **Hướng dẫn kịch bản Test Case kiểm tra bảo mật (Securty Testing)**
* **Security là gì?**  
  Security là các biện pháp được thiết lập để bảo vệ một ứng dụng chống lại các hành động không lường trước được rằng các hành động đó sẽ ảnh hưởng hoặc phá hủy ứng dụng. Hành động không lường trước có thể là cố ý hoặc vô ý.
* **Security testing là gì?**  
  Security Testing là việc tìm kiếm tất cả cá lỗ hổng có thể và điểm yếu của hệ thống mà có thể dẫn đến mất thông tin trong tay nhân viên hoặc người ngoài của tổ chức. Security Testing rất quan trọng trong ngành công nghiệp CNTT để bảo vệ dữ liệu của tất cả các phương tiện.
* **Mục đích của Security Testing?**  
  Mục đích của Security Testing là để xác định các mối đe dọa trong hệ thống và đo lường rủi ro tiềm năng của nó. Nó cũng giúp trong việc phát hiện các nguy cơ bảo mật có thể có trong hệ thống và giúp các developer trong việc sửa chữa những vấn đề này thông qua code
* **Kịch bản**

1. Mật khẩu phải ở trong định dạng mã hóa
2. Gõ trực tiếp url vào thanh địa chỉ của trình duyệt mà không qua đăng nhập. Các trang nội bộ phải không được mở.
3. Sau khi đăng nhập và mở các trang nội bộ, thay đổi url trực tiếp bằng cách đổi tham số ID của trang tới trang thuộc quyền người dùng đã đăng nhập khác. Truy cập phải bị từ chối bởi người dùng này không thể xem trang thống kê của người dùng khác.
4. Thử các giá trị đầu vào không hợp lệ trong các trường username, password. Hệ thống phải báo lỗi.
5. Các thư mục web, các tệp tin không được truy nhập trực tiếp mà không có tùy chọn “Download”.
6. Kiểm tra CAPTCHA cho các đăng nhập tự động
7. Tất cả các phiên giao dịch, các thông báo lỗi, các hành vi cố gắng xâm phạm an ninh phải ghi trong log và lưu tại web server.
8. Kiểm tra thời gian cookie và session cho ứng dụng
9. Đối với các trang web tài chính, nút Back của Browser lại không nên được hoạt động.

* **Các trường hợp kiểm tra để kiểm tra bảo mật:**

1. Cố gắng truy cập trực tiếp vào trang web được đánh dấu mà không cần đăng nhập vào hệ thống.
2. Xác minh rằng hệ thống sẽ hạn chế bạn tải xuống tệp mà không cần đăng nhập vào hệ thống.
3. Xác minh rằng các trang được truy cập trước đó không thể truy cập được sau khi đăng xuất tức là Đăng xuất và sau đó nhấn nút Quay lại để truy cập trang đã truy cập trước đó.
4. Kiểm tra mật khẩu hợp lệ và không hợp lệ, quy tắc mật khẩu cho biết không thể dưới 6 ký tự, id người dùng và mật khẩu không thể giống nhau, v.v.
5. Đã xác minh rằng thông tin quan trọng tức là nhạy cảm như mật khẩu, số ID, số thẻ tín dụng, v.v. sẽ không được hiển thị trong hộp nhập khi nhập. Chúng nên được mã hóa và ở định dạng dấu hoa thị.
6. Kiểm tra đánh dấu trang có bị vô hiệu hóa trên các trang an toàn không? Đánh dấu trang nên được vô hiệu hóa trên các trang an toàn.
7. Kiểm tra là Nhấp chuột phải, Xem, Nguồn bị vô hiệu hóa? Mã nguồn không được hiển thị cho người dùng.
8. Có cách nào khác để truy cập các trang bảo mật cho các trình duyệt trong phiên bản không?
9. Kiểm tra xem máy chủ của bạn có khóa một cá nhân đã cố gắng truy cập trang web của bạn nhiều lần với thông tin đăng nhập / mật khẩu không hợp lệ không?
10. Xác minh điều kiện hết thời gian chờ, sau khi hết thời gian sử dụng, người dùng sẽ không thể điều hướng qua trang web.
11. Kiểm tra Bạn có bị ngăn thực hiện tìm kiếm trực tiếp bằng cách chỉnh sửa nội dung trong URL không?
12. Xác minh rằng thông tin liên quan nên được ghi vào tệp nhật ký và thông tin đó phải được theo dõi.
13. Trong SSL xác minh rằng mã hóa được thực hiện chính xác và kiểm tra tính toàn vẹn của thông tin.
14. Xác minh rằng người dùng không thể truy cập trang bị hạn chế sau khi hết thời gian phiên.
15. Xác thực ID / mật khẩu, cùng một tài khoản trên các máy khác nhau không thể đăng nhập cùng một lúc. Vì vậy, tại một thời điểm chỉ có một người dùng có thể đăng nhập vào hệ thống bằng id người dùng.
16. Phương thức xác thực ID / mật khẩu đã nhập sai mật khẩu nhiều lần và kiểm tra xem tài khoản có bị khóa hay không.
17. Thêm hoặc sửa đổi thông tin quan trọng (mật khẩu, số ID, số thẻ tín dụng, v.v.). Kiểm tra nếu nó được phản ánh ngay lập tức hoặc lưu trữ các giá trị cũ.
18. Xác minh rằng Thông báo lỗi không chứa thông tin độc hại để tin tặc sẽ sử dụng thông tin này để hack trang web.

* **Hướng dẫn kịch bản Test Case kiểm tra hiệu suất (Performance Testing)**
* Thời gian phản hồi: xác định xem ứng dụng phản hồi nhanh hay chậm
* Khả năng mở rộng: Xác định tải người dùng tối đa mà ứng dụng phần mềm có thể xử lý.
* Tính ổn định: Xác định xem ứng dụng có ổn định dưới các tải khác nhau hay không.
* **Bảng so sánh Performance Test, Load Test và Stress Test**

|  |  |  |
| --- | --- | --- |
| **Performance Test** | **Load Test** | **Stress Test** |
| Bao gồm cả Load Test và Stress Test | Là một loại của Performance Test | Là một loại của Performance Test |
| Giúp tạo ra thiết lập chuẩn và tiêu chuẩn cho ứng dụng | Giúp nhận ra giới hạn của hệ thống, thiết lập SLA của ứng dụng và kiểm tra hệ thống có khả năng chịu tải như thế nào | Kiểm tra xem hệ thống hoạt động như thế nào khi quá tải và cách hệ thống phục hồi khi xảy ra lỗi |
| Mục đích của Performance Test là tạo ra hướng dẫn về cách hệ thống hoạt động khi ở điều kiện bình thường | Tạo ra những kịch bản khi hệ thống hoạt động quá tải | Stress Test nhằm đảm bảo rằng khi hoạt động trong điều kiện tải cao trong một khoảng thời gian cố định sẽ không bị crash |
| Việc sử dụng tài nguyên, khả năng đáp ứng và độ tin cậy của sản phẩm được kiểm tra ở phương pháp kiểm thử này | Các thuộc tính được kiểm tra trong một bài kiểm tra tải là hiệu suất hoạt động lúc cao điểm, số lượng máy chủ và thời gian phản hồi. | Loại kiểm thử này kiểm tra thời gian phản hồi ổn định, v.v. |
| Trong Performance Test, giới hạn tải bao gồm cả dưới và trên ngưỡng nghỉ. | Trong Load Test giới hạn tải là ngưỡng ngắt. | Trong Stress Test giới hạn tải là trên ngưỡng nghỉ. |
| Ví dụ: Kiểm tra nhiều người dùng cùng một thời điểm, kết nối HTTP hoặc kiểm tra thời gian phản hồi thích hợp. | Ví dụ : Kiểm tra trình xử lý từ bằng cách thay đổi một phần lớn data, kiểm tra máy in bằng cách truyền dữ liệu nặng. Kiểm tra máy chủ mail với hàng ngàn người dùng đồng thời. | Ví dụ : Đột nhiên tắt và khởi động lại các port của một mạng lưới lớn. |
| **Tại sao thực hiện Performance Test?** | **Tại sao thực hiện Load Test?** | **Tại sao thực hiện Stress Test?** |
| Để kiểm tra xem ứng dụng đang hoạt động chính xác hay không | Tìm ra lỗi mà không thể tìm ra với bất kỳ phương pháp thử nghiệm khác. Chẳng hạn như rò rỉ bộ nhớ, lỗi quản lý bộ nhớ, tràn bộ đệm, v.v ... | Nó giúp các đơn vị kiểm tra hệ thống khi xảy ra lỗi. |
| Để phù hợp với nhu cầu hoạt động của doanh nghiệp | Để phù hợp với nhu cầu hoạt động của doanh nghiệp | Để đảm bảo rằng hệ thống có sao lưu dữ liệu trước khi xảy ra lỗi hay không |
| Tìm, phân tích và khắc phục các vấn đề về hiệu suất | Để xác định độ ổn định của một ứng dụng | Để kiểm tra xem bất kì trục trắc nào làm ảnh hưởng xấu đến an ninh hệ thống hay không |
| Xác định tất cả phần cứng để đưa ra dự kiến tải phù hợp. | Để kiểm tra xem cơ sở hạ tầng hiện tại có đủ để chạy ứng dụng hay không. |  |
| Thực hiện kế hoạch về nhu cầu đáp ứng năng lực trong tương lai của ứng dụng | Số lượng người dùng đồng thời mà một ứng dụng có thể hỗ trợ và khả năng mở rộng để cho phép nhiều người dùng truy cập vào nó |  |

* **Quy trình kiểm thử hiệu năng cơ bản sẽ diễn ra với các bước sau:**
* ***Bước 1: Xác định môi trường kiểm thử***

Môi trường kiểm thử bao gồm các yếu tố: Môi trường kiểm thử vật lý cùng môi trường sản xuất cũng như các công cụ kiểm tra có sẵn. Ngoài ra, yếu tố cấu hình của phần cứng, phần mềm và internet dùng cho quá trình kiểm thử cũng được quan tâm. Đây là tất cả các yếu tố quyết định quá trình kiểm thử hiệu quả nhất và xác định được những khó khăn sẽ gặp phải trong quá trình kiểm thử hiệu năng sản phẩm.

* ***Bước 2: Xác định các tiêu chí về tính chấp nhận hiệu năng có thể chấp nhận được của hệ thống.***

Các yếu tố cần quan tâm gồm có: Các mục tiêu, ràng buộc cho thông lượng, yếu tố phân bổ nguồn lực và thời gian phản hồi. Tại giai đoạn này, người tester cần phải xác định được những tiêu chí cùng mục tiêu về hiệu năng tối thiểu cần đạt của hệ thống. Điều này có ý nghĩa quan trọng trong công đoạn thiết lập các tiêu chí về hiệu năng.

* ***Bước 3: Lập kế hoạch, thiết kế kiểm thử hiệu năng***

Ở công đoạn này, ta cần phải xác định được các cách sử dụng ở những khả năng khác nhau giữa các người dùng cuối đồng thời xác định những tình huống chính nhằm giúp cho việc kiểm tra toàn bộ các trường hợp sử dụng có thể. Đây là quá trình cần thiết nhằm mô phỏng lại một loạt người dùng cuối cũng như kế hoạch về dữ liệu thử nghiệm hiệu xuất, sau cùng là nhằm phác thảo lại các số liệu khi thu thập.

* **Bước 4: Cài đặt môi trường cho kiểm thử**

Trước khi tiến hành Performance testing, cần phải sẵn sàng về việc cài đặt môi trường, tài nguyên và các công cụ cần thiết.

* ***Bước 5: Tiến hành test design***

Đây là quá trình tạo các trường hợp test để kiểm thử hiệu năng theo test design.

* ***Bước 6: Thực hiện test***

Ngoài thực thi thì cần phải theo dõi kết quả của quá trình thực hiện test.

* ***Bước 7: Phân tích, điều chỉnh và kiểm tra lại***

Đây là khâu quan trọng nhằm tinh chỉnh, kiểm tra lại xem có cần điều chỉnh về hiệu suất (tăng tốc độ xử lý CPU) hay không.

* **Các trường hợp sử dụng Load Test**

1. Tải xuống hàng loạt file lớn từ internet.
2. Chạy đồng thời nhiều ứng dụng trên máy tính hoặc máy chủ.
3. Để nhiều công việc cho một máy in trong một hàng đợi.
4. Đang chịu một máy chủ lưu lượng truy cập lớn.
5. Viết và đọc dữ liệu đến và đi từ đĩa cứng liên tục.