

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN

ĐINH THỊ LAN ANH

**TÌM HIỂU CÔNG CỤ SELENIUM WEBDRIVER
VÀ ỨNG DỤNG KIỂM THỬ WEBSITE
CUNGDOCSACHVOITOI.COM**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM**

Thái Nguyên - 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM

Đề Tài

TÌM HIỂU CÔNG CỤ SELENIUM WEBDRIVER
VÀ ỨNG DỤNG KIỂM THỬ WEBSITE
CUNGDOCSACHVOITOI.COM

Sinh viên thực hiện: Đinh Thị Lan Anh

Lớp/Khóa: KTPM K18A

Giáo viên hướng dẫn: Đinh Khánh Linh

Thái Nguyên - 2024

LỜI CẢM ƠN

Trong quá trình học tập và làm Đồ án tốt nghiệp em đã rất may mắn nhận được sự giúp đỡ tạo điều kiện của các thầy cô trong Trường ĐH Công Nghệ Thông Tin và Truyền Thông - ĐH Thái Nguyên nói chung và bộ môn Công nghệ phần mềm nói riêng. Em xin gửi lời cảm ơn chân thành tới giáo viên hướng dẫn là Đinh Khánh Linh cùng các thầy, cô giáo trong bộ môn Công nghệ phần mềm đã tận tình hướng dẫn em; cảm ơn các anh chị làm việc trong đơn vị ETS tại công ty ETS Software đã nhiệt tình giúp đỡ em hoàn thành Đồ án tốt nghiệp theo đúng thời gian quy định.

Do thời gian và kinh nghiệm thực tế còn hạn chế nên có những phần thực hiện chưa được tốt và khó tránh khỏi thiếu sót, em rất mong nhận được sự góp ý, thông cảm và chỉ bảo tận tình của Thầy cô và các bạn cho bài báo cáo này được hoàn chỉnh hơn nữa và đạt kết quả cao.

Em xin chân thành cảm ơn!

Thái Nguyên, ngày 04 tháng 12 năm 2023

Sinh viên thực hiện

Đinh Thị Lan Anh

MỤC LỤC

LỜI CAM ĐOAN	6
CHƯƠNG 1: TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM	7
1.1 Kiểm thử phần mềm là gì	7
1.2 Những nguyên tắc của kiểm thử phần mềm	7
1.2.1. Phân tích yêu cầu	7
1.2.2. Lập kế hoạch kiểm thử (Test Plan).....	8
1.2.3. Thiết kế và tạo testcase	8
1.2.4. Cài đặt môi trường test	9
1.2.5. Thực hiện kiểm thử (Test Executetion)	9
1.2.6. Giai đoạn đóng quy trình (Test cycal Closure)	9
1.3 Những nguyên tắc của kiểm thử	10
1.3.1. Kiểm thử đưa ra lỗi	10
1.3.2. Kiểm thử cạn kiệt là không thể.....	10
1.3.3. Kiểm thử càng sớm càng tốt	10
1.3.4. Sự tập trung của lỗi.....	11
1.3.5. Nghịch lí thuốc trừ sâu	11
1.3.6. Kiểm thử phụ thuộc vào ngữ cảnh.....	11
1.3.7. Không có lỗi - Sai lầm	12
1.4 Phân loại kiểm thử.....	13
1.4.1 Kiểm thử tĩnh (Static Testing):	13
1.4.2 Kiểm thử động (Dynamic Testing):	13
1.5 Kiểm thử thủ công và kiểm thử tự động.	18
1.5.1 Kiểm thử thủ công	18
1.5.2 Kiểm thử tự động.....	19
CHƯƠNG 2: KIỂM THỬ TỰ ĐỘNG.....	24
2.1 Giới thiệu bộ công cụ Selenium webdriver	25
2.2 Selenium WebDriver.....	28

2.2.1 Lý do tại sao chúng ta nên sử dụng Selenium WebDriver:	28
2.2.2 Cài đặt Selenium WebDriver.....	29
2.3 Các câu lệnh thường dùng khi làm việc với Selenium WebDriver	33
2.3.1 Các câu lệnh trên trình duyệt:.....	33
2.3.2 Các câu lệnh định vị phần tử GUI	36
2.3.3 Các câu lệnh Wait.....	40
2.3.4 Câu lệnh PageLoadTimeout	41
2.3.5 Câu lệnh SetScriptTimeout.....	42
2.3.6.Câu lệnh Sleep	42
2.4 TestNG Framework.....	42
2.4.1 TestNG là gì.....	42
2.5 Find Web Element.....	46
2.6 POM.....	48
CHƯƠNG 3: ÁP DỤNG KIỂM THỬ WEBSITE CUNGDOCSACHVOITOI.COM	51
3.1 Mô tả Website CUNGDOCSACHVOITOI.COM	51
3.2 Kế hoạch kiểm thử	52
3.3 Kiểm thử thủ công.....	53
3.3.1 Thiết kế kịch bản kiểm thử auto	53
3.3.2 Báo cáo kiểm thử thủ công	58
3.4 Kiểm thử tự động với Selenium WebDriver	61
3.4.1 Xây dựng testscript dựa trên kịch bản thủ công	61
3.4.2 Thực thi testscript	66
3.4.3 Phân tích kết quả.....	67
3.5 Kiểm thử giao diện thủ công	70
3.5.1 Thiết kế kịch bản kiểm thử giao diện thủ công	70
3.5.2 Báo cáo kết quả kiểm thử giao diện thủ công.....	72
3.6 Đánh giá kiểm thử thủ công và kiểm thử tự động.....	73
KẾT LUẬN	76

LỜI CAM ĐOAN

Em xin cam đoan nội dung trong bài báo cáo thực tập tốt nghiệp này là kết quả của quá trình nghiên cứu tìm hiểu thực sự của cá nhân em, được thực hiện dựa trên cơ sở lý thuyết và sự hướng dẫn nhiệt tình chỉ bảo của giảng viên Đinh Khánh Linh

Thái Nguyên, ngày 04 tháng 12 năm 2023

Sinh viên thực hiện

Đinh Thị Lan Anh

CHƯƠNG 1: TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM

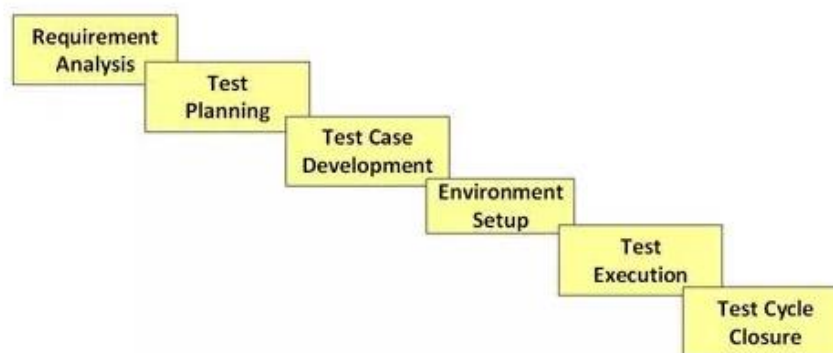
1.1 Kiểm thử phần mềm là gì

Kiểm thử phần mềm là quá trình kiểm tra và đánh giá một phần mềm để đảm bảo rằng nó hoạt động như mong đợi và đáp ứng các yêu cầu được xác định trước. Quá trình này bao gồm kiểm tra tính đúng đắn, hiệu suất, độ tin cậy và tính bảo mật của phần mềm. Mục tiêu của kiểm thử phần mềm là phát hiện và sửa chữa các lỗi, đảm bảo chất lượng sản phẩm cuối cùng, cũng như tăng sự tin cậy và hài lòng của người dùng.

Quá trình này thường được thực hiện bởi các chuyên gia kiểm thử hoặc nhóm kiểm thử dưới sự hướng dẫn của các kỹ sư phần mềm. Kiểm thử có thể cung cấp cho doanh nghiệp một quan điểm, một cách nhìn độc lập về phần mềm để từ đó cho phép đánh giá và thấu hiểu được những rủi ro trong quá trình triển khai phần mềm.[1]

1.2 Những nguyên tắc của kiểm thử phần mềm

Quy trình kiểm thử phần mềm là một chuỗi các bước được thực hiện để đảm bảo chất lượng của sản phẩm phần mềm. Nó gồm nhiều phương pháp và hoạt động nhằm xác minh rằng phần mềm hoạt động như mong đợi và đáp ứng được các yêu cầu cụ thể. Mặc dù có nhiều phương pháp kiểm thử khác nhau được áp dụng trên toàn cầu, nhưng vẫn chưa có một tiêu chuẩn chung được chấp nhận rộng rãi để định rõ chu trình kiểm thử phần mềm.



Hình 1: Quy trình kiểm thử phần mềm

Các giai đoạn của kiểm thử phần mềm:

1.2.1. Phân tích yêu cầu

Nhóm kiểm thử sẽ tiến hành phân tích tài liệu yêu cầu để có cái nhìn tổng quan về phần mềm, từ đó có thể định rõ các yêu cầu cần được kiểm tra. Các thành viên trong nhóm

kiểm thử/ QA có thể liên hệ với các bên liên quan như khách hàng, BA (Business Analyst), Leader, PM, ... để hiểu rõ hơn về hệ thống thông qua cuộc trò chuyện trực tiếp hoặc việc tạo file câu hỏi và trả lời.

Các yêu cầu ở đây có thể là các yêu cầu về chức năng (xác định những gì phần mềm cần thực hiện) hoặc là các yêu cầu phi chức năng (về hiệu suất, bảo mật, ...).

Trong giai đoạn này, cũng sẽ đánh giá tính khả thi của việc tự động hóa kiểm thử. Nhiệm vụ bao gồm việc xác định loại kiểm thử cần thực hiện, thu thập thông tin về mức độ ưu tiên của kiểm thử (đặc biệt là xác định chức năng nào là quan trọng nhất và cần được kiểm tra trước), xác định môi trường kiểm thử và phân tích khả thi của việc tự động hóa kiểm thử (nếu dự án yêu cầu).[1]

1.2.2. Lập kế hoạch kiểm thử (Test Plan)



Hình 2: Kế hoạch kiểm thử

Test plan là một tài liệu tổng quan về quá trình kiểm thử của một dự án, mô tả rõ ràng về phạm vi của dự án, cách tiếp cận, quy trình kiểm thử, tài nguyên và nhân lực cần thiết, các tính năng được kiểm tra và không được kiểm tra, cũng như các công cụ và môi trường kiểm thử cần sử dụng. Đây là cơ sở để thực hiện kiểm thử cho các sản phẩm hoặc phần mềm trong dự án.

Trong giai đoạn này, Test Manager hoặc Test Leader sẽ phát triển kế hoạch ban đầu về kiểm thử. Kế hoạch kiểm thử sẽ bao gồm các thông tin sau: xác định phạm vi kiểm thử, chiến lược kiểm thử, danh sách các tài liệu liên quan như spec, tài liệu thiết kế, các kế hoạch kiểm thử khác, các vấn đề ưu tiên và mục tiêu kiểm thử, ước lượng thời gian và thời gian thực tế cho kiểm thử, ước lượng số lượng testcase cần tạo cho mỗi chức năng hoặc mỗi module.

Nhiệm vụ trong giai đoạn này bao gồm chuẩn bị kế hoạch kiểm thử và tài liệu chiến lược cho các loại kiểm thử khác nhau, lựa chọn các công cụ kiểm thử, và ước lượng thời gian cần thiết để thực hiện kiểm thử.

1.2.3. Thiết kế và tạo testcase

Trong bước này, những người kiểm thử sẽ đảm nhận việc lập kế hoạch kiểm thử bằng cách tạo testcase hoặc checklist (với kiểm thử thủ công) và script kiểm thử (với kiểm thử tự động) dựa trên tài liệu yêu cầu. Họ sẽ áp dụng các kỹ thuật thiết kế testcase để thực hiện

công việc này. Đồng thời, việc chuẩn bị dữ liệu kiểm thử cũng sẽ được thực hiện trong giai đoạn này.

Nhiệm vụ bao gồm tạo ra các checklist/testcase hoặc script kiểm thử, đồng thời đánh giá lại chúng để đảm bảo rằng chúng phản ánh đúng yêu cầu đã được đặc tả. Cũng trong phạm vi nhiệm vụ này là việc tạo dữ liệu kiểm thử phù hợp.

1.2.4. Cài đặt môi trường test

Môi trường kiểm thử đề cập đến cấu hình phần mềm và phần cứng được sử dụng để thực hiện các testcase. Đây là một phần quan trọng trong quy trình kiểm thử và thường được tiến hành đồng thời với quá trình thiết kế testcase. Trong một số trường hợp, nhóm tester/ QA có thể không cần thực hiện bước này nếu môi trường kiểm thử đã được cung cấp sẵn bởi khách hàng hoặc nhóm phát triển.

Nhiệm vụ bao gồm hiểu rõ kiến trúc của hệ thống, thiết lập môi trường kiểm thử, chuẩn bị phần cứng và phần mềm cần thiết, cài đặt môi trường và dữ liệu kiểm thử.

1.2.5. Thực hiện kiểm thử (Test Execution)



Hình 3: Thực thi kiểm thử

Trong bước này, các thành viên kiểm thử sẽ tiến hành thực hiện các bước kiểm thử dựa trên kế hoạch và các kịch bản đã được chuẩn bị trước đó. Mọi lỗi phát hiện sẽ được ghi lại và báo cáo cho nhóm phát triển phần mềm để tiến hành sửa chữa.

Nhiệm vụ bao gồm thực hiện các bước kiểm thử theo kế hoạch, ghi lại lỗi và các thông tin liên quan khi các testcase không thành công, thực hiện kiểm tra lại sau khi các lỗi đã được sửa, và theo dõi quá trình sửa lỗi và đóng các lỗi đã được giải quyết.

1.2.6. Giai đoạn đóng quy trình (Test Cycle Closure)

Nhóm kiểm thử sẽ tổ chức cuộc họp để thảo luận và phân tích các tài liệu (như yêu cầu, tài liệu thiết kế, testcase, kế hoạch kiểm thử, kịch bản kiểm thử, ...) để đề xuất các chiến lược tương lai và rút ra bài học từ quá trình kiểm thử hiện tại. Mục đích của hoạt

động này là giải quyết các khó khăn xuất hiện trong dự án và tìm ra các phương pháp tối ưu cho các dự án trong tương lai.

Nhiệm vụ bao gồm đánh giá tiêu chí hoàn thành kiểm thử dựa trên thời gian và kết quả test, chi phí và chất lượng. Đồng thời, chuẩn bị báo cáo kết quả kiểm thử, bao gồm cả báo cáo định tính và định lượng về sản phẩm cho khách hàng. Cuối cùng, phân tích kết quả kiểm thử để xác định sự phân phối lỗi theo loại và mức độ nghiêm trọng.

1.3 Những nguyên tắc của kiểm thử

1.3.1. Kiểm thử đưa ra lỗi

Nguyên tắc này đặt ra yêu cầu bắt đầu thực hiện các hoạt động kiểm thử phần mềm từ giai đoạn đầu của quy trình phát triển phần mềm. Việc này giúp phát hiện sớm các lỗi và vấn đề trong phần mềm. Điều này làm cho việc chuyển giao phần mềm theo yêu cầu, đúng thời gian và chất lượng dự kiến, trở nên khả thi hơn.

Kiểm thử có thể cho thấy rằng phần mềm đang có lỗi, nhưng không thể chứng minh rằng phần mềm không có lỗi. Kiểm thử làm giảm xác suất lỗi chưa tìm thấy vẫn còn trong phần mềm, nhưng ngay cả khi đã kiểm thử nghiêm ngặt phần mềm vẫn có thể còn lỗi.

1.3.2. Kiểm thử cạn kiệt là không thể

Kiểm thử cạn kiệt là một nguyên tắc trong kiểm thử phần mềm. Nguyên tắc này đề xuất rằng không thể kiểm thử mọi khả năng, mọi trường hợp hoặc mọi điều kiện có thể xảy ra trong quá trình sử dụng phần mềm. Thay vào đó, nhóm kiểm thử cần tập trung vào việc tối ưu hóa các tài nguyên kiểm thử của họ để đạt được phạm vi kiểm thử rộng và đa dạng nhất có thể.

Việc kiểm thử cạn kiệt hoàn toàn sẽ tốn kém về thời gian, công sức và nguồn lực, trong khi vẫn không đảm bảo rằng mọi lỗi sẽ được phát hiện. Thay vào đó, các nhóm kiểm thử thường sử dụng các kỹ thuật như kiểm thử kiểu mẫu, kiểm thử tương đương và phân tích nguyên nhân cơ bản để tìm ra các trường hợp kiểm thử quan trọng và tiềm năng nhất.

Mục tiêu của kiểm thử không phải là kiểm thử mọi khả năng có thể, mà là tìm ra càng nhiều lỗi càng tốt trong một nguồn lực có hạn. Điều này cần một cân bằng giữa việc đảm bảo rằng các trường hợp quan trọng được kiểm thử và việc không vô ích bỏ thời gian vào việc kiểm thử các trường hợp ít quan trọng hoặc không thực tế.

1.3.3. Kiểm thử càng sớm càng tốt

Nguyên tắc này yêu cầu bắt đầu thử nghiệm phần mềm trong giai đoạn đầu của vòng đời phát triển phần mềm. Các hoạt động kiểm thử phần mềm từ giai đoạn đầu sẽ giúp phát hiện bug sớm hơn. Nó cho phép chuyển giao phần mềm theo yêu cầu đúng thời gian với chất lượng dự kiến.

Phát hiện lỗi sớm: Bằng cách kiểm thử từ giai đoạn sớm của quy trình phát triển, các lỗi và vấn đề có thể được phát hiện và khắc phục ngay từ đầu, giúp giảm thiểu chi phí và thời gian cần thiết để sửa chữa sau này.