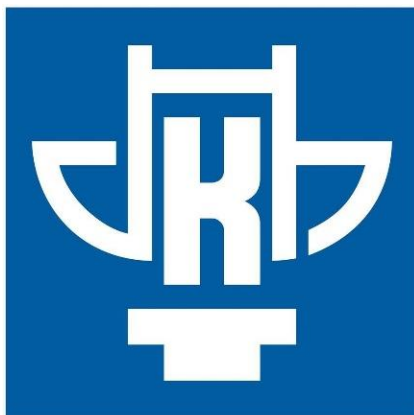


Trường Đại Học Kiến Trúc Hà Nội
Khoa Công Nghệ Thông Tin



**Môn học: Kiểm Thử Và Đảm Bảo Chất
Lượng Phần Mềm**

**ĐỀ TÀI: SỬ DỤNG SELENIUM IDE ĐỂ KIỂM THỬ
NHỮNG CHỨC NĂNG CƠ BẢN WEBSITE SHOPEE**

Giảng viên hướng dẫn: TS. Nguyễn Đăng Khoa
Lớp: 20CN4
Nhóm 2: Vũ Nguyễn Tiến Lộc

Hà Nội, 03/2024

MỤC LỤC

I. PHẦN MỀM VÀ KIỂM THỬ PHẦN MỀM	3
1.1. Phần mềm và khái niệm liên quan	3
1.1.1. Phần mềm	3
1.1.2. Lỗi phần mềm	3
1.1.3. Yêu cầu của khách hàng	3
1.1.4. Đặc tả yêu cầu phần mềm	4
1.1.5. Chất lượng và độ tin cậy của phần mềm	5
1.2. Kiểm thử phần mềm	5
1.2.1. Khái niệm	5
1.2.2. Vai trò của kiểm thử phần mềm	5
1.2.3. Các cấp độ trong kiểm thử phần mềm	6
1.2.4. Quy trình kiểm thử phần mềm	7
1.2.5. Phân loại kiểm thử phần mềm	9
1.2.6. Các mức độ nghiêm trọng của lỗi	11
1.2.7. Ca kiểm thử	12
1.2.8. Kiểm thử tự động	14
1.2.9. Nguyên tắc quan trọng trong kiểm thử phần mềm	15
1.3. Các kỹ thuật xác định ca kiểm thử	16
1.3.1. Kỹ thuật phân vùng tương đương	17
1.3.2. Kỹ thuật phân tích giá trị biên	17
1.3.3. Đoán lỗi	18
1.3.4. Kỹ thuật chuyển trạng thái	19
II. KIỂM THỬ ỨNG DỤNG TRÊN NỀN WEB BẰNG CÔNG CỤ SELENIUM	19
2.1. Công cụ kiểm thử tự động Selenium	19
2.1.1. Giới thiệu chung về Selenium	19
2.1.2. Selenium IDE (Selenium Integrated Development Environment)	20
2.1.3. Selenium RC (Selenium Remote Control)	21
2.1.4. Selenium WebDriver	24
2.1.5. Selenium Grid	25
III. SỬ DỤNG SELENIUM IDE ĐỂ KIỂM THỬ NHỮNG CHỨC NĂNG CƠ BẢN WEBSITE SHOPEE	25

3.1. Các module kiểm thử	25
3.2. Xây dựng các module kiểm thử	25
3.2.1. Module 1 - Kiểm tra phản hồi khi đăng nhập tài khoản	25
3.2.2. Module 2 - Kiểm tra phản hồi khi quên mật khẩu	27
3.2.3. Module 3 - Kiểm tra phản hồi khi tìm kiếm sản phẩm	28
3.2.4. Module 4 - Kiểm tra phản hồi khi mua hàng	29
3.2.5. Module 5 - Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng	29
3.2.6. Module 6 - Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng	30
3.2.7. Module 7 - Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm	30
3.2.8. Module 8 - Kiểm tra phản hồi khi đăng ký tài khoản mới	31
IV. TÀI LIỆU THAM KHẢO	31

MỤC LỤC HÌNH ẢNH

Hình 1: Vòng đời của quá trình kiểm thử	6
Hình 2: Quy trình kiểm thử phần mềm.	7
Hình 3: Xác định ca kiểm thử với kiểm thử hộp trắng	10
Hình 4: Minh họa kỹ thuật kiểm thử hộp đen.	11
Hình 5: Biểu đồ Venn nguồn các ca kiểm thử	11
Hình 6: Bảng phân loại mức độ nghiêm trọng của lỗi.	12
Hình 7: Minh họa mẫu ca kiểm thử đơn giản	13
Hình 8: Minh họa kỹ thuật phân vùng tương đương	17
Hình 9: Minh họa kỹ thuật phân tích giá trị biên.	18
Hình 10: Ưu nhược điểm của Selenium IDE.	20
Hình 11: Minh họa test ứng dụng nào mà sử dụng Selenium Core.	22
Hình 12: Hỗ trợ từ Selenium RC	22
Hình 13: Ưu nhược điểm của Selenium RC	23
Hình 14: Module 1 - Kiểm tra phản hồi khi đăng nhập tài khoản	26
Hình 15: Module 2 - Kiểm tra phản hồi khi quên mật khẩu.	27
Hình 16: Module 3 - Kiểm tra phản hồi khi tìm kiếm sản phẩm.	28
Hình 17: Module 4 - Kiểm tra phản hồi khi mua hàng.	29
Hình 18: Module 5 - Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng.	29
Hình 19: Module 6 - Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng.	30
Hình 20: Module 7 - Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm.	30
Hình 21: Module 8 - Kiểm tra phản hồi khi đăng ký tài khoản mới.	31

I. PHẦN MỀM VÀ KIỂM THỬ PHẦN MỀM

1.1. Phần mềm và khái niệm liên quan

1.1.1. Phần mềm

Phần mềm thường được mô tả với các bộ phận cấu thành:

- Tập các lệnh (chương trình máy tính) trên máy tính khi được thực hiện sẽ tạo ra các dịch vụ và đem lại những kết quả mong muốn cho người dùng.
- Các cấu trúc dữ liệu và giải thuật (lưu giữ trên các bộ nhớ) làm cho chương trình thao tác hiệu quả với các thông tin thích hợp và nội dung thông tin được số hoá.
- Các tài liệu để mô tả các thao tác, cách sử dụng và bảo trì phần mềm (hướng dẫn sử dụng, tài liệu kỹ thuật, tài liệu phân tích, thiết kế, kiểm thử, v.v.).
- Dữ liệu cần thiết cho sự vận hành của hệ thống (có thể kể đến như tham số đầu vào, mã nguồn, danh sách tên thích hợp với đặc tả).

1.1.2. Lỗi phần mềm

Lỗi phần mềm nhìn chung là sự không khớp giữa chương trình và đặc tả của nó, kéo theo những vấn đề xuất hiện trong các giai đoạn phát triển phần mềm.

Lỗi phần mềm thường xuất hiện ở các hình thức sau đây:

- Sai (Fault): Khi phần mềm gặp lỗi sẽ đưa đến những sai sót. Tuy nhiên, không dễ để phát hiện ra sai sót trong quá trình phát triển phần mềm. Sai lầm có thể xuất hiện ở ngay đầu quy trình phát triển phần mềm khi người phân tích, thiết kế bỏ sót thông tin dẫn tới thiếu chức năng mà lẽ ra cần phải có.
- Thất bại (Failure): Thất bại dễ nhận thấy nhất khi một lỗi được thực thi. Chúng thường xuất hiện dưới 2 dạng: thất bại có thể chạy được (ví dụ như mã nguồn) và thất bại chỉ liên kết với các lỗi về nhiệm vụ. Ngoài ra, có thể kể đến các thất bại liên quan tới các lỗi do bỏ quên. Chúng ta có thể hạn chế thất bại ngay tại bước đầu tiên của quy trình phát triển phần mềm nếu việc khảo sát được thực hiện tốt.
- Sự cố (Incident): Sự cố thường được liên kết với một thất bại. Tuy nhiên nó khác với thất bại ở chỗ sự cố luôn hiển thị cho người dùng hoặc kiểm thử viên biết về sự tồn tại của nó.
- Thừa: 1 số chức năng không có trong bản đặc tả yêu cầu phần mềm nhưng lại xuất hiện trong phần mềm được xây dựng.

Ngoài ra, còn xuất hiện 1 số lỗi phi chức năng như phần mềm khó sử dụng, tốc độ không đáp ứng yêu cầu (vấn đề hiệu năng) hay giao diện khó nhìn cũng dễ khiến cho người sử dụng nghĩ rằng phần mềm đang hoạt động không đúng.

1.1.3. Yêu cầu của khách hàng

Phần mềm được phát triển dựa trên nhu cầu của khách hàng. Chính vì lẽ đó, các chức năng của phần mềm được xây dựng dựa trên việc thu thập, phân tích, khảo sát nhu cầu của khách hàng thông qua những yêu cầu cụ thể. Đối với phần mềm, yêu cầu thường

được tổng hợp từ nhiều người, nhiều tổ chức có mức độ chuyên môn và mức độ tham gia cũng như tương tác với phần mềm khác nhau trong môi trường hoạt động của nó.

Có thể phân loại yêu cầu của khách hàng cho sản phẩm phần mềm thành một số loại như sau:

Phân loại theo sản phẩm và tiến trình

- Yêu cầu sản phẩm: là những đòi hỏi hay ràng buộc mà phần mềm phải thực hiện .
- Yêu cầu tiến trình: là những ràng buộc liên quan đến việc phát triển phần mềm (kỹ thuật sử dụng, mô hình phát triển, v.v.).

Ví dụ: Khách hàng muốn phát triển một website làm bài thi trực tuyến. Lúc này, yêu cầu sản phẩm là xây dựng website thi trực tuyến với các tính năng như quản lý câu hỏi; quản lý đề thi; cho phép người dùng có thể tham gia làm bài thi; quản trị viên có thể duyệt các câu hỏi và bộ đề thi trước khi đăng lên website. Việc website được phát triển theo mô hình Agile hay mô hình thác nước chính là yêu cầu tiến trình của sản phẩm phần mềm.

Phân loại theo chức năng

- Yêu cầu chức năng: đặc tả các chức năng mà phần mềm cần phải thực hiện.
- Yêu cầu phi chức năng: là các ràng buộc về giải pháp và chất lượng (hiệu năng, việc bảo trì, mức độ an toàn, bảo mật, v.v.).
- Yêu cầu đặc tả các thuộc tính nổi bật: là đặc tả cho các thuộc tính phụ thuộc vào sự vận hành, đặc biệt là kiến trúc hệ thống. Các thuộc tính này không thể xác định được cho từng thành phần đơn lẻ.

Phân loại theo tính kiểm định

- Những yêu cầu mang tính mơ hồ, không thể kiểm định
- Những yêu cầu đã rõ ràng và có thể kiểm định được.

Phân loại theo phạm vi đặc tả

- Yêu cầu hệ thống: đặc tả các cấu hình, cơ sở hạ tầng, phần cứng, phần mềm, con người, kỹ thuật, v.v. của toàn bộ hệ thống.
- Yêu cầu phần mềm: đặc tả các chức năng, giao diện, v.v. của các cấu phần phần mềm.

1.1.4. Đặc tả yêu cầu phần mềm

Từ yêu cầu của khách hàng và những yêu cầu bắt buộc khác, đặc tả yêu cầu phần mềm được viết ra để mô tả một cách chính xác các yêu cầu cần đáp ứng của sản phẩm phần mềm. Đây cũng chính là tài liệu cơ sở để lập trình viên, kiểm thử viên và các bộ phận khác dựa vào để phát triển phần mềm hoàn chỉnh, đúng với yêu cầu đặt ra ban đầu. Các khái niệm về lỗi đã nói ở mục 1.1.2 cũng chính là đề cập đến việc phần mềm sau khi xây dựng

hoạt động không đúng với bản đặc tả yêu cầu phần mềm. Tài liệu đặc tả yêu cầu phần mềm cũng cần cung cấp đầy đủ các thông tin về chi phí, rủi ro và lịch trình cho quá trình phát triển sản phẩm.

Đặc tả yêu cầu phần mềm được viết ra phục vụ rất nhiều đối tượng từ người dùng hệ thống, khách hàng đến các nhà phát triển và bảo trì phần mềm. Do đó, tài liệu đặc tả nên được viết bằng ngôn ngữ tự nhiên, sử dụng biểu đồ, bảng biểu để đảm bảo tính dễ hiểu, dễ sử dụng cho tất cả các đối tượng trên.

1.1.5. Chất lượng và độ tin cậy của phần mềm

Chất lượng của phần mềm trước hết là sự đáp ứng các yêu cầu đề ra trong bản đặc tả yêu cầu phần mềm. Có thể kể đến các yếu tố đại diện cho chất lượng phần mềm như: tính đúng đắn, tính hiệu quả, độ tin cậy, tính khả kiểm thử, dễ học, dễ sử dụng, dễ bảo trì... Ta có thể thấy độ tin cậy chỉ là một trong những yếu tố đánh giá chất lượng phần mềm. Tuy nhiên người kiểm thử lại hay nhầm lẫn giữa khái niệm chất lượng và độ tin cậy của phần mềm. Sau quá trình kiểm thử đảm bảo phần mềm có thể chạy ổn định, kiểm thử viên thường sẽ cho rằng phần mềm lúc này đã đạt chất lượng tốt.

Độ tin cậy của phần mềm là xác suất để phần mềm chạy không có thất bại trong một khoảng thời gian nhất định. Ngoài ra, có thể dựa vào thời gian khắc phục sự cố để đánh giá độ tin cậy của phần mềm.

Trong phần tiếp theo, chúng ta sẽ tìm hiểu về khái niệm cũng như các vấn đề xung quanh việc Kiểm thử phần mềm.

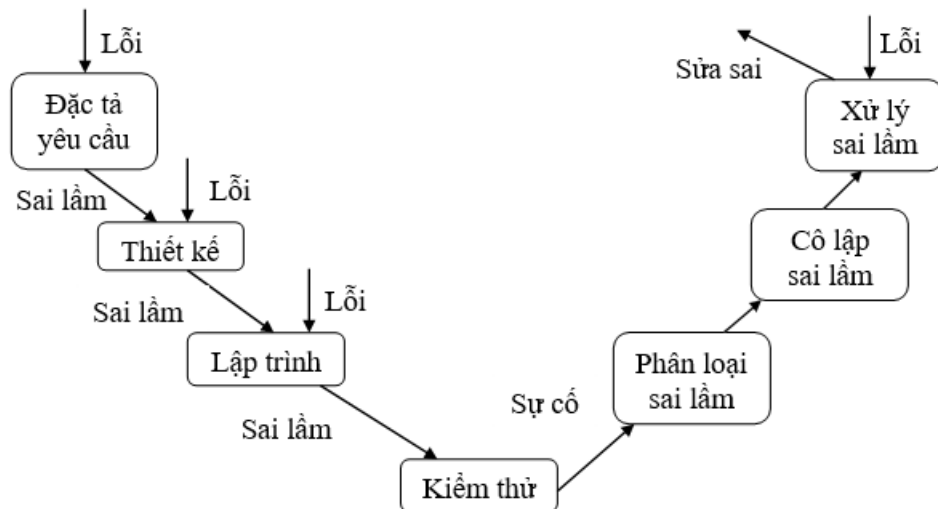
1.2. Kiểm thử phần mềm

1.2.1. Khái niệm

Kiểm thử phần mềm là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng của sản phẩm hoặc dịch vụ được kiểm thử. Hiểu theo cách đơn giản hơn, kiểm thử phần mềm là quá trình tìm thất bại hoặc chứng tỏ việc tiến hành của phần mềm là đúng đắn.

1.2.2. Vai trò của kiểm thử phần mềm

Kiểm thử phần mềm chiếm một vị trí quan trọng trong việc nâng cao chất lượng cũng như độ tin cậy của phần mềm trong quá trình phát triển. Hoàn thành vòng quay “đưa lỗi vào – tìm lỗi – khử lỗi đi” của quy trình kiểm thử phần mềm sẽ thu lại được những cải tiến đáng kể cho chất lượng sản phẩm phần mềm. Việc biết được sản phẩm phần mềm tốt tới mức nào trước khi đưa vào sử dụng sẽ hạn chế tối đa những rủi ro gặp phải trong quá trình phát triển phần mềm.



Hình 1: Vòng đời của quá trình kiểm thử.

1.2.3. Các cấp độ trong kiểm thử phần mềm

Có rất nhiều cách để chia cấp độ kiểm thử phần mềm, nhưng tựu chung lại sẽ gồm 4 cấp độ sau:

Kiểm thử đơn vị: Cấp độ này chủ yếu do lập trình viên trực tiếp thực hiện. Phần mềm khi phát triển sẽ bao gồm nhiều đơn vị chức năng (hàm, phương thức) hợp thành. Mỗi lập trình viên sẽ đảm nhiệm việc phát triển một hay nhiều đơn vị chức năng. Kiểm thử đơn vị chính là việc lập trình viên sau khi hoàn thành code đơn vị chức năng của mình sẽ tiến hành kiểm thử chức năng đó một cách cô lập nhằm phát hiện ra lỗi và khắc phục trước khi tích hợp với các đơn vị chức năng khác. Kiểm thử đơn vị thường được tiến hành theo 2 giai đoạn: kiểm thử đơn vị tĩnh và kiểm thử đơn vị động.

Kiểm thử tích hợp: Sau khi kiểm thử đơn vị được tiến hành bởi chính lập trình viên viết ra nó, các đơn vị chức năng sẽ được ghép lại với nhau để tạo thành hệ thống đầy đủ và có thể làm việc được. Các đơn vị chức năng hoạt động tốt khi ở trạng thái độc lập riêng rẽ, nhưng khi ghép lại sẽ có thể xuất hiện những lỗi về giao diện hoặc cho ra kết quả không đúng khi phải sử dụng dữ liệu từ những đơn vị chức năng khác. Đó chính là lý do tại sao phải tiếp tục kiểm thử để phát hiện ra những lỗi kể trên. Người ta thường chia bước kế tiếp này thành 2 giai đoạn: kiểm thử tích hợp và kiểm thử hệ thống. Ở mức kiểm thử tích hợp, các đơn vị chức năng được kết hợp lại với nhau và tiến hành kiểm thử chúng theo phương pháp tăng dần để đảm bảo cụm các đơn vị chức năng sẽ làm việc ổn định trong môi trường thử nghiệm.

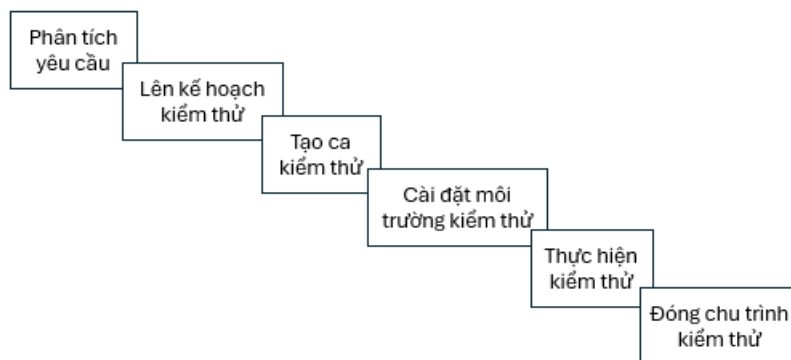
Kiểm thử hệ thống: Sau khi tất cả các đơn vị chức năng đã được tích hợp lại với nhau tạo thành một hệ thống hoàn chỉnh, kiểm thử hệ thống sẽ được thực thi để đảm bảo sản phẩm phần mềm đáp ứng đầy đủ các yêu cầu trong bản đặc tả yêu cầu phần mềm. Đây là công việc tốn nhiều công sức nhất trong quá trình kiểm thử phần mềm. Đồng thời cũng

sử dụng nhiều kỹ thuật kiểm thử khác nhau như: kiểm thử giao diện người dùng, kiểm thử chức năng, kiểm thử hiệu năng, kiểm thử tính dễ dùng, v.v. để hoàn tất công việc kiểm thử trong cấp độ này.

Kiểm thử chấp nhận: Khi kiểm thử hệ thống hoàn tất, sản phẩm phần mềm được coi như đã sẵn sàng cho việc đưa vào sử dụng thực tế. Lúc này, phần mềm cần được tiến hành cấp độ kiểm thử cuối cùng – kiểm thử chấp nhận bởi chính khách hàng hay người sử dụng phần mềm. Tuy có phần tương tự như kiểm thử hệ thống nhưng mục đích chính của kiểm thử chấp nhận là quyết định việc đưa vào sử dụng chính thức sản phẩm phần mềm. Người ta dựa trên các số liệu thống kê thực tế về chất lượng, độ tin cậy của phần mềm để quyết định triển khai cho người dùng cuối. Kiểm thử chấp nhận thường được thực hiện dưới hình thức cho một nhóm người dùng thử sản phẩm phần mềm để phát hiện các lỗi và nhận phản hồi từ người dùng. Trong đó, phiên bản alpha dành cho đội phát triển phần mềm và phiên bản beta được cung cấp cho người sử dụng thật để đưa ra đánh giá trong môi trường thực tế. Ở thời điểm hiện tại, kiểm thử chấp nhận được coi là cấp độ quy chuẩn bắt buộc không thể thiếu trong quy trình phát triển của nhiều sản phẩm phần mềm.

1.2.4. Quy trình kiểm thử phần mềm

Kiểm thử phần mềm bao gồm nhiều giai đoạn với sự phối hợp của nhiều bên liên quan chứ không chỉ là một hoạt động đơn lẻ. Chính vì thế, cần có quy trình kiểm thử phần mềm để làm rõ các công đoạn, các bước kiểm thử, người chịu trách nhiệm và khi nào việc kiểm thử được tiến hành trong toàn bộ quy trình phát triển phần mềm. Nói cách khác, quy trình kiểm thử phần mềm chính là chuỗi các hoạt động được tiến hành để thực hiện việc kiểm thử. Các giai đoạn trong quy trình kiểm thử phần mềm được biểu diễn tổng quát bằng sơ đồ sau:



Hình 2: Quy trình kiểm thử phần mềm.

Phân tích yêu cầu: Nhóm kiểm thử sẽ tương tác với các bên liên quan để hiểu rõ những yêu cầu cụ thể cần cho việc kiểm thử. Các yêu cầu có thể là chức năng (xác định phần mềm cần phải làm những gì) hoặc phi chức năng (hiệu năng, tính bảo mật hệ thống, màu sắc, v.v.)

- Hoạt động cụ thể:
 - Xác định loại kiểm thử sẽ thực hiện.
 - Tổng hợp chi tiết về và mức độ tập trung thử tự ưu tiên.
 - Chuẩn bị RTM (Requirement Traceability Matrix – một tài liệu dưới dạng bảng sử dụng để theo dõi các yêu cầu của khách hàng và kiểm tra xem các yêu cầu này đã được đáp ứng đầy đủ hay chưa)
 - Xác định môi trường kiểm thử.
 - Phân tích khả năng sử dụng kiểm thử tự động.
- Tài liệu sử dụng:
 - RTM.
 - Báo cáo về khả năng sử dụng kiểm thử tự động (nếu cần).

Lên kế hoạch kiểm thử: Còn được gọi bằng tên khác là lên chiến lược thử nghiệm. Ở giai đoạn này, trưởng nhóm kiểm thử sẽ dự toán chi phí cho dự án cũng như chuẩn bị kế hoạch kiểm thử.

- Hoạt động cụ thể:
 - Lựa chọn công cụ kiểm thử (test tool).
 - Lên kế hoạch về nhân sự và ấn định vai trò trách nhiệm cho từng người trong nhóm.
 - Phổ biến cho mọi người trong nhóm kiểm thử về yêu cầu dự án.
- Tài liệu sử dụng:
 - Bản kế hoạch kiểm thử.

Tạo ca kiểm thử: Giai đoạn này cần phải tạo, xác minh, kiểm tra lại các ca kiểm thử. Dữ liệu kiểm thử cũng được tạo và xác định trong giai đoạn này.

- Hoạt động cụ thể:
 - Tạo ca kiểm thử.
 - Xác minh, kiểm tra lại các ca kiểm thử.
 - Tạo dữ liệu kiểm thử.
- Tài liệu sử dụng:
 - Ca kiểm thử.
 - Dữ liệu kiểm thử.

Cài đặt môi trường kiểm thử: Môi trường kiểm thử quyết định bởi các điều kiện phần cứng và phần mềm trong từng dự án. Thiết lập môi trường kiểm thử có thể thực hiện song song với giai đoạn sinh ca kiểm thử và là một tiêu chí quan trọng trong quá trình kiểm thử. Tuy nhiên, nhóm kiểm thử có thể không cần tham gia vào giai đoạn này nếu đã có các bên

liên quan khác hỗ trợ, nhiệm vụ của nhóm kiểm thử chỉ là yêu cầu môi trường kiểm thử cần thiết.

- Hoạt động cụ thể:
 - Hiểu được kiến trúc yêu cầu, thiết lập môi trường và chuẩn bị danh sách yêu cầu về phần cứng và phần mềm cho môi trường thử nghiệm.
 - Thiết lập môi trường kiểm thử.

Thực hiện kiểm thử: Nhóm kiểm thử thực hiện kiểm thử theo kế hoạch và danh sách ca kiểm thử đã chuẩn bị từ giai đoạn trước. Các lỗi phát hiện ở giai đoạn này sẽ được thông báo lại cho nhóm phát triển phần mềm để chỉnh sửa và thực hiện kiểm thử lại.

- Hoạt động cụ thể:
 - Thực hiện kiểm thử theo kế hoạch.
 - Làm tài liệu về kết quả kiểm thử, cập nhật lại các lỗi trong ca kiểm thử.
 - Kiểm thử lại các lỗi đã được chỉnh sửa.
 - Kiểm tra để đóng lỗi.
- Tài liệu sử dụng:
 - Ca kiểm thử (cập nhật kết quả).
 - Báo cáo lỗi.

Đóng chu trình kiểm thử: Nhóm kiểm thử sẽ họp, thảo luận và phân tích những bài học rút ra sau quá trình kiểm thử, đưa ra chiến lược cho những lần kiểm thử kế tiếp hoặc chia sẻ kinh nghiệm cho những dự án tương tự.

- Hoạt động cụ thể:
 - Đánh giá việc hoàn thành quy trình kiểm thử dựa vào thời gian, mức độ bao phủ, chi phí và chất lượng.
 - Chuẩn bị dữ liệu dựa trên các tiêu chí trên.
 - Chuẩn bị báo cáo kết thúc kiểm thử.
 - Báo cáo chất lượng sản phẩm cho khách hàng.
 - Phân tích kết quả kiểm thử để tìm ra sự phân bố lỗi theo loại và mức độ nghiêm trọng.
- Tài liệu sử dụng:
 - Báo cáo kết thúc kiểm thử.

1.2.5. Phân loại kiểm thử phần mềm

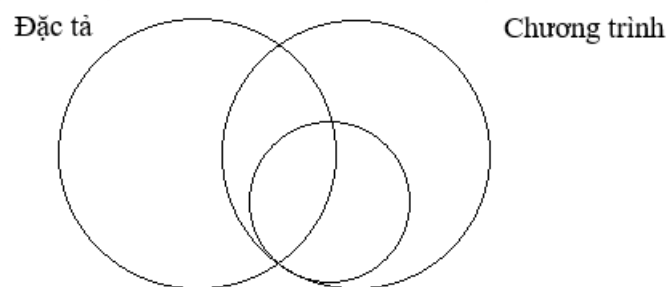
Có 2 cách cơ bản để xác định các ca kiểm thử là kiểm thử tĩnh và kiểm thử động.

Kiểm thử tĩnh: là một hình thức của kiểm thử phần mềm mà không cần thực thi chương trình. Điều này ngược với thử nghiệm động. Công việc chủ yếu là kiểm tra tính đúng đắn của mã lệnh, thuật toán hay tài liệu. Đây là loại kiểm thử được thực hiện bởi lập trình viên.

Lỗi được phát hiện bằng kiểm thử tĩnh ít tốn kém để sửa chữa hơn so với lỗi phát hiện bằng kiểm thử động sẽ được đề cập dưới đây. Các lập trình viên có thể trao đổi mã nguồn chéo nhau hoặc làm việc một cách độc lập để thực hiện kiểm thử tĩnh.

Kiểm thử động: Liên quan đến việc thực thi chương trình để phát hiện các lỗi, thất bại có thể có của chương trình hay tìm ra các vấn đề về hiệu năng hệ thống. Việc thực thi chương trình trên tất cả các dữ liệu đầu vào là không thể nên ta chỉ có thể chọn một tập con các dữ liệu đầu vào để thực thi hay nói cách khác là sinh ra các ca kiểm thử. Trong kiểm thử động, người ta chia làm 2 kỹ thuật: kiểm thử hộp trắng (kiểm thử cấu trúc) và kiểm thử hộp đen (kiểm thử chức năng).

- **Kiểm thử hộp trắng:** là kỹ thuật kiểm thử dựa vào thuật toán, cấu trúc mã nguồn bên trong của chương trình với mục đích đảm bảo rằng tất cả các câu lệnh và điều kiện sẽ được thực hiện ít nhất một lần. Người kiểm thử truy cập vào mã nguồn chương trình và kiểm tra nó, lấy đó làm cơ sở để thực hiện việc kiểm thử. Kiểm thử hộp trắng bao gồm các công việc cơ bản: Kiểm thử đường dẫn, kiểm thử luồng điều khiển, kiểm thử nội bộ (xác nhận các tham số, vòng lặp), kiểm thử tính năng (kiểm tra thời gian xử lý, dữ liệu cụ thể). Tuy nhiên, việc kiểm thử hộp trắng tồn tại khá nhiều hạn chế như: không thể đảm bảo rằng chương trình đã tuân theo đặc tả, khó phát hiện được các lỗi do dữ liệu, thiếu đường dẫn, v.v. Như vậy, không thể chỉ sử dụng kiểm thử hộp trắng để kiểm thử chương trình.



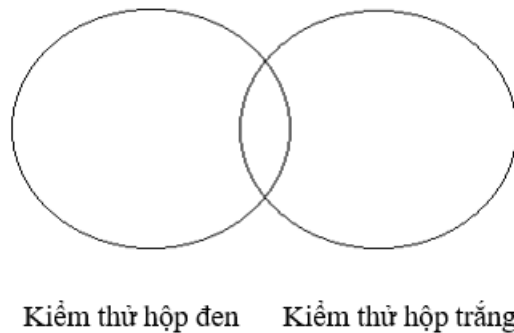
Hình 3: Xác định ca kiểm thử với kiểm thử hộp trắng.

- **Kiểm thử hộp đen:** Là kỹ thuật kiểm thử dựa trên đầu vào và đầu ra của chương trình mà không quan tâm tới mã nguồn bên trong được viết ra sao. Với kỹ thuật này, kiểm thử viên xem phần mềm như là một hộp đen. Để thực hiện, kiểm thử viên sẽ xây dựng các nhóm giá trị đầu vào sao cho chúng có thể thực hiện đầy đủ các chức năng cần có của chương trình. Kiểm thử hộp đen sử dụng các phương pháp: phân tích giá trị biên, kiểm thử tính bền vững, kiểm thử trường hợp xấu nhất, kiểm thử phân lớp tương đương miền dữ liệu đầu vào, đầu ra, kiểm thử giá trị đặc biệt, kiểm thử dựa trên bảng quyết định. Tất cả các phương pháp trên đều dựa trên thông tin xác định về các thành phần đang được kiểm thử.



Hình 4: Minh hoạ kỹ thuật kiểm thử hộp đen.

Cho tới nay, việc xác định kỹ thuật kiểm thử nào là tốt hơn trong 2 kỹ thuật: kiểm thử hộp đen và kiểm thử hộp trắng vẫn còn là một dấu hỏi lớn. Biểu đồ Venn sau đây sẽ giúp hình dung khái quát về mối liên hệ giữa kiểm thử hộp đen và kiểm thử hộp trắng trong thực tế kiểm thử hiện nay.



Hình 5: Biểu đồ Venn nguồn các ca kiểm thử.

Trước hết cần khẳng định mục đích của hai kỹ thuật trên đều là để xác định ca kiểm thử. Trong khi kiểm thử hộp trắng chỉ dùng đặc tả để xác định ca kiểm thử, thì kiểm thử hộp đen lại dùng mã nguồn chương trình để làm cơ sở xác định ca kiểm thử. Trong phần trình bày chi tiết về hai kỹ thuật đã nói ở trên đều cho thấy không có kỹ thuật nào là đủ tốt hoàn toàn. Cụ thể: Nếu tất cả các hành vi được nêu trong bản đặc tả yêu cầu phần mềm vẫn chưa được cài đặt thì kiểm thử hộp trắng sẽ không thể nhận biết được điều đó. Hay nếu các hành vi đã được cài đặt trong chương trình nhưng lại chưa có trong bản đặc tả yêu cầu phần mềm, kiểm thử hộp đen dường như bất lực trong trường hợp này. Qua biểu đồ Venn, ta có thể khẳng định: việc kết hợp khéo léo cả hai kỹ thuật kiểm thử trên sẽ đem lại một kết quả tốt nhất trong kiểm thử phần mềm. Thực hiện song song kiểm thử hộp đen và kiểm thử hộp trắng sẽ hạn chế tối đa các khiếm khuyết có thể bị bỏ sót. Tuy nhiên, nếu biết được loại lỗi nào hay mắc phải hoặc những sai lầm thường thấy trong dự án đang được triển khai, ta hoàn toàn có thể chọn kỹ thuật kiểm thử thích hợp cho từng ca kiểm thử. Kinh nghiệm của kiểm thử viên sẽ được phát huy trong những trường hợp này.

1.2.6. Các mức độ nghiêm trọng của lỗi

Chương trình một khi đã xuất hiện lỗi đều kéo theo những hệ lụy nghiêm trọng. Một trong những cách phân loại mức độ nghiêm trọng của lỗi thường được sử dụng là dựa trên

tần suất xuất hiện: chỉ một lần, thỉnh thoảng, xuất hiện lại hay lặp đi lặp lại nhiều lần. Việc phân loại mức độ nghiêm trọng của lỗi sẽ giúp kiểm thử viên cũng như lập trình viên ý thức được đâu là lỗi cần được giải quyết trước, nhằm giảm thiểu tối đa những tổn thất về chi phí và nâng cao chất lượng cho sản phẩm phần mềm. Hình 1.6 dưới đây minh họa các mức độ nghiêm trọng của lỗi dựa trên độ nghiêm trọng và hậu quả.

1	Nhẹ	Lỗi chính tả
2	Vừa	Hiểu lầm hoặc thừa thông tin
3	Khó chịu	Tên bị thiếu, cắt chữ hoặc hoá đơn có giá trị 0.0 đồng
4	Bực mình	Một vài giao dịch không được xử lý
5	Nghiêm trọng	Mất giao dịch
6	Rất nghiêm trọng	Xử lý giao dịch sai
7	Cực kỳ nghiêm trọng	Lỗi rất nghiêm trọng thường xuyên xảy ra
8	Quá quắt	Hủy hoại cơ sở dữ liệu
9	Thảm họa	Hệ thống dừng hoạt động
10	Dịch họa	Thảm họa chuyển sang mức lây lan

Hình 6: Bảng phân loại mức độ nghiêm trọng của lỗi.

1.2.7. Ca kiểm thử

Ca kiểm thử là một khái niệm không thể thiếu trong kiểm thử phần mềm. Theo ISTQB “ca kiểm thử là một tập hợp các giá trị đầu vào, tiền điều kiện, các kết quả mong đợi và điều kiện kết thúc, được xây dựng cho mục đích hoặc điều kiện kiểm thử riêng biệt để kiểm tra tính đúng đắn của chương trình với yêu cầu của bản đặc tả yêu cầu phần mềm” [5]. Hay nói cách khác, ca kiểm thử mô tả dữ liệu bao gồm: đầu vào, hành động hoặc sự kiện và kết quả đầu ra mong đợi (expected results) để xác định liệu 1 ứng dụng, hệ thống phần mềm hoặc một trong các tính năng của nó có hoạt động đúng như mong muốn hay không.

Cấu trúc của một ca kiểm thử thông thường bao gồm:

- Test case ID: Xác định số lượng trường hợp cần kiểm thử.
- Function (Chức năng): Các function có thể được chia nhỏ dựa theo chức năng của hệ thống nhằm giúp ca kiểm thử trở nên rõ ràng hơn.
- Pre-condition: Điều kiện đầu vào của ca kiểm thử, ví dụ như khi thực hiện kiểm thử form đăng nhập, pre-condition sẽ là form đăng nhập phải được hiển thị ra.
- Test Data: Dữ liệu đầu vào cần chuẩn bị trước khi kiểm thử.
- Test Steps: Mô tả chi tiết các bước thực hiện kiểm thử.
- Expected Results: Kết quả mong đợi sau khi thực hiện các bước kiểm thử.

- Actual result: Mô tả kết quả thực tế khi thực hiện kiểm thử trên môi trường của hệ thống. Actual result thường bao gồm ba giá trị: pass, fail và pending.
- Comments: Có thể chứa screen shot hoặc thông tin liên quan khi thực hiện ca kiểm thử.

Ngoài ra có thể có thêm một số cột như: Designed by (người thực hiện kiểm thử), Execute Date (ngày thực hiện kiểm thử), v.v. Mức độ chi tiết của ca kiểm thử sẽ phụ thuộc vào từng dự án và quy mô của công ty sản xuất phần mềm.

Test Case #: 2.2 System: ATM Designed by: ABC Executed by: Short Description: Test the ATM Change PIN service	Test Case Name: Change PIN Subsystem: PIN Design Date: 28/11/2004 Execution Date:	Page: 1 of 1
--	--	---------------------

Pre-conditions The user has a valid ATM card - The user has accessed the ATM by placing his ATM card in the machine The current PIN is 1234 The system displays the main menu

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click the 'Change PIN' button	The system displays a message asking the user to enter the new PIN		
2	Enter '5555'	The system displays a message asking the user to confirm (re-enter) the new PIN		
3	Re-enter '5555'	The system displays a message of successful operation The system asks the user if he wants to perform other operations		
4	Click 'YES' button	The system displays the main menu		
5	Check post-condition 1			

Post-conditions 1. The new PIN '5555' is saved in the database
--

Hình 7: Minh họa mẫu ca kiểm thử đơn giản.

Một ca kiểm thử được cho là hiệu quả khi:

- Dựa vào ca kiểm thử có thể tìm thấy lỗi.
- Tìm được nhiều lỗi khó phát hiện.
- Chỉ ra được những điểm ban đầu mà khi thực hiện kiểm thử không tìm ra vấn đề.
- Ca kiểm thử cần có những bước thực hiện kiểm thử (Test steps) đơn giản, minh bạch, dễ hiểu.
- Các trường hợp thử nghiệm nên có giá trị, tóm tắt và ngắn.
- Các ca kiểm thử nên có sự liên kết: Mỗi ca kiểm thử cần được đánh số thứ tự (Test case ID) để đảm bảo ca kiểm thử đã bao phủ 100% bản đặc tả yêu cầu phần mềm.
- Ca kiểm thử có thể bảo trì: Nên viết ca kiểm thử sao cho khi có thay đổi, chỉnh sửa thì các bên liên quan có thể dễ dàng nhận thấy được sự thay đổi đó.
- Ca kiểm thử có tính ứng dụng cao.

Tóm lại, ca kiểm thử được viết ra để kiểm tra hoạt động của các chức năng có đúng như mong muốn trong bản đặc tả yêu cầu phần mềm hay không. Khi viết ca kiểm thử nên cố gắng viết đơn giản, dễ hiểu nhưng phải đầy đủ các dữ liệu chuẩn cần có của một ca kiểm thử.

1.2.8. Kiểm thử tự động

Kiểm thử tự động là quá trình kiểm tra một hệ thống nào đó bằng các công cụ tự động hoá với dữ liệu đầu vào và đầu ra đã được xác định.

Công việc kiểm thử thường chiếm từ 11% - 40% chi phí cho quá trình phát triển phần mềm. Hơn nữa, các dự án phần mềm đều mong muốn giảm chi phí về thời gian, nhân lực mà vẫn đem lại hiệu quả cao, chất lượng tốt. Đó chính là lý do kiểm thử tự động được áp dụng rộng rãi trong các quy trình phát triển phần mềm ngày nay.

Kiểm thử tự động đặc biệt phát huy tác dụng trong các trường hợp kiểm thử lặp đi lặp lại, kiểm thử hồi quy hay các ca kiểm thử có giá trị dữ liệu đầu vào rất lớn khiến cho việc kiểm thử thủ công gặp nhiều khó khăn. Đối với các trường hợp kiểm thử lặp đi lặp lại, nếu thực hiện thủ công sẽ gây ra sự nhàm chán cho người kiểm thử, dẫn tới năng suất lao động kém. Đó là chưa kể tới việc lặp đi lặp lại quy trình một cách thủ công hoàn toàn có thể dẫn tới sai sót. Ngược lại, nếu thay bằng kiểm thử tự động, dù có lặp đi lặp lại bao nhiêu lần thì cũng cho ra thao tác và kết quả chính xác. Điều này giúp chúng ta tránh được những rủi ro không đáng có và giảm đáng kể thời gian cho việc kiểm thử.

Một số công cụ kiểm thử tự động phổ biến hiện nay:

- Selenium: Công cụ kiểm thử phần mềm mã nguồn mở hỗ trợ kiểm thử tự động cho các ứng dụng Web. Selenium cung cấp chức năng ghi tự động và phát lại, hỗ trợ hữu ích cho kiểm thử hồi quy. Điểm mạnh của Selenium là hỗ trợ nhiều nền tảng khác nhau, tích hợp vào các trình duyệt phổ biến hiện nay, có thể thực hiện nhiều ca kiểm thử cùng lúc, có khả năng lưu các ca kiểm thử để sử dụng lại khi cần và cho phép người dùng chèn chú thích ở giữa kịch bản kiểm thử để hiểu rõ hơn nội dung kiểm thử. Selenium cũng hỗ trợ một lượng lớn các ngôn ngữ lập trình Web phổ biến hiện nay như C#, Java, Perl, PHP, Python, Ruby, v.v. Selenium có thể kết hợp với một số công cụ khác như Bromien và Junit nhưng với người dùng thông thường chỉ cần chạy tự động mà không cần cài thêm các công cụ hỗ trợ. Selenium hiện nay đang được cộng đồng đánh giá là một trong những công cụ tốt nhất cho kiểm thử tự động các ứng dụng Web.
- QTP (HP UFT): Được sử dụng rộng rãi để kiểm thử chức năng và hồi quy. QTP sử dụng khái niệm kiểm thử từ khoá để đơn giản hoá việc tạo và bảo trì ca kiểm thử. Công cụ này hỗ trợ môi trường .NET và có cơ chế xác định đối tượng kiểm thử tốt.

Đối với những kiểm thử viên không theo ngành kỹ thuật vẫn có thể sử dụng dễ dàng công cụ này.

- Rational Function Tester: Là 1 công cụ kiểm tra tự động hướng đối tượng có khả năng tự động kiểm tra dữ liệu, kiểm tra giao diện, và kiểm thử hồi quy. Rational Function Tester hỗ trợ rất nhiều giao thức và ứng dụng như Java, HTML, .NET, Windows, Visual Basic, v.v. Công cụ này cũng hỗ trợ ghi và phát lại các hành động theo yêu cầu. Nó cho phép các nhà phát triển tạo ra các kịch bản liên quan đến từ khóa để có thể được tái sử dụng. Bộ biên tập Công cụ Java Developer Toolkit của Eclipse tạo điều kiện cho kiểm thử viên tạo mã thử nghiệm các đoạn mã trong Java với Eclipse.
- WATIR: Là một phần mềm kiểm tra mã nguồn mở để kiểm thử hồi quy. Watir hỗ trợ nhiều trình duyệt trên nhiều nền tảng khác nhau. Đồng thời cũng sử dụng một ngôn ngữ kịch bản hiện đại có đầy đủ các tính năng. Điểm mạnh của Watir là hỗ trợ ứng dụng Web được viết bởi bất kỳ ngôn ngữ nào.
- SilkTest: Silk Test được thiết kế để thực hiện kiểm thử chức năng và hồi quy. Nó là một ngôn ngữ hướng đối tượng giống như C++. SilkTest sử dụng các khái niệm về đối tượng, các class và sự kế thừa. Nó chuyển đổi các lệnh script thành các lệnh GUI. Trên cùng một máy, các lệnh có thể được chạy trên một máy từ xa hoặc máy chủ. Silktest có thể xác định chuyển động của con chuột cùng với các phím bấm. Nó có thể sử dụng cả phương pháp ghi và phát lại hoặc các phương pháp lập trình mô tả.

Dù có rất nhiều ưu điểm về mặt thời gian thực thi nhưng kiểm thử tự động cũng không thể thay thế hoàn toàn quá trình kiểm thử của con người. Để thực hiện kiểm thử tự động, trước hết vẫn cần bàn tay con người thiết lập thao tác cho công cụ hay các đoạn kịch bản máy tính để thực thi. Đối với những ca kiểm thử chỉ thực hiện số ít lần thì việc mất thời gian tạo kịch bản kiểm thử tự động là không cần thiết. Chưa kể tới những ca kiểm thử với đặc thù riêng biệt mà kiểm thử tự động không làm được. Thêm vào đó, không phải công cụ kiểm thử tự động nào cũng miễn phí và dễ sử dụng hay đưa vào triển khai rộng rãi.

1.2.9. Nguyên tắc quan trọng trong kiểm thử phần mềm

Có thể hiểu nguyên tắc là những quy định mà chúng ta phải tuân theo. Trong kiểm thử phần mềm, việc theo đuổi những nguyên tắc là điều cần thiết giúp chúng ta phát triển hệ thống một cách tốt nhất. Người ta đưa ra 7 nguyên tắc kiểm thử quan trọng có thể dựa vào chúng để tiết kiệm thời gian, công sức và chi phí phát triển. Cụ thể:

1. Kiểm thử chỉ ra lỗi: Kiểm thử có thể phát hiện ra lỗi của phần mềm, nhưng lại không thể chứng minh phần mềm hoàn toàn không có lỗi. Thực tế cho thấy, ngay cả khi kiểm thử một cách nghiêm ngặt phần mềm vẫn có thể xuất hiện lỗi. Vì vậy, cần phải tìm được ra càng nhiều lỗi càng tốt.

2. Kiểm thử cạn kiệt là không thể: Đó chính là việc chúng ta không thể kiểm tra với mọi dữ liệu đầu vào, đầu ra với toàn bộ các kịch bản của một phần mềm. Hay nói cách khác, kiểm tra mọi thứ trong chương trình một cách trọn vẹn là điều không thể làm được. Tuy nhiên, có thể phân tích rủi ro và dựa trên mức độ ưu tiên thông qua các mức độ nghiêm trọng của lỗi phần mềm để kiểm thử một số chức năng chính, thiết yếu của phần mềm.
3. Kiểm thử càng sớm càng tốt: Chi phí cho việc khắc phục, sửa lỗi sẽ tỷ lệ thuận với thời gian phát hiện ra lỗi đó. Không ít phần mềm khi chuẩn bị giao cho khách hàng mới phát hiện ra lỗi xuất hiện ngay từ bản đặc tả yêu cầu phần mềm hay bản phân tích thiết kế hệ thống. Điều này gây ra những thiệt hại không nhỏ cho quá trình phát triển phần mềm. Vì vậy, kiểm thử phần mềm ngay từ những giai đoạn đầu của quy trình phát triển là điều hết sức cần thiết.
4. Xác định vị trí tập trung lỗi: Lỗi thường tập trung nhiều ở những chức năng chính của phần mềm. Do đó, tập trung tìm ra lỗi ở những chức năng này sẽ giúp giảm thiểu thời gian kiểm thử. Đây là một trong những cách cơ bản và hiệu quả nhất trong kiểm thử phần mềm.
5. Nghịch lý thuốc trừ sâu: Dữ liệu của một hệ thống thay đổi liên tục và thường xuyên xuất hiện những hành vi mới từ phía người dùng. Nó giống như việc sử dụng một loại thuốc trừ sâu trong nhiều mùa vụ mà không để ý tới sự phát triển đa dạng của sâu bọ. Vì vậy, nếu giữ nguyên ca kiểm thử cũ trong thời gian dài, chúng ta không thể tìm ra lỗi mới phát sinh. Việc xem xét và thay đổi các ca kiểm thử thường xuyên là điều cần thiết.
6. Kiểm thử phụ thuộc vào ngữ cảnh: Nguyên tắc này đề cập tới việc tiếp cận kiểm thử theo nhiều ngữ cảnh khác nhau. Điển hình như việc kiểm thử ứng dụng Web và kiểm thử trên thiết bị di động không thể dùng chung một chiến lược kiểm thử. Mỗi môi trường kiểm thử sẽ có những đặc trưng riêng và đó là lý do tại sao cần xác định việc kiểm thử theo ngữ cảnh cho phù hợp.
7. Phần mềm có lỗi bằng 0: Việc không tìm thấy lỗi không có nghĩa là không tồn tại lỗi trong phần mềm. Chúng ta chỉ có thể hạn chế tối đa lỗi có thể gặp phải chứ không thể triệt tiêu toàn bộ lỗi có thể gặp phải.

Dựa theo 7 nguyên tắc trên đây giúp chúng ta có cái nhìn tổng quát về kiểm thử phần mềm cũng như đánh giá được tính hiệu quả của hoạt động kiểm thử đang triển khai.

1.3. Các kỹ thuật xác định ca kiểm thử

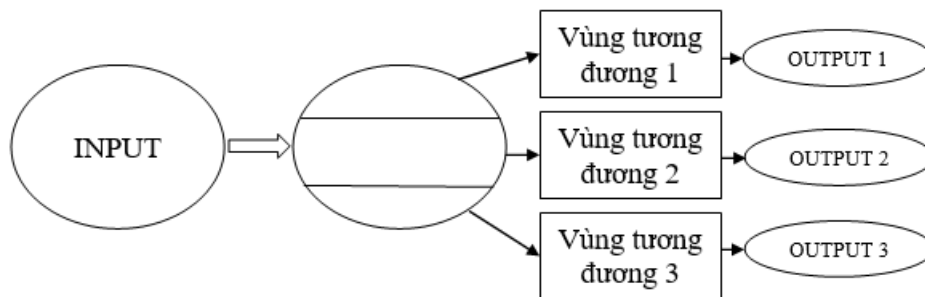
Trong quá trình kiểm thử phần mềm sẽ nảy sinh vô số trường hợp cần phải xét tới. Tuy nhiên, vì yếu tố chi phí, thời gian phát triển dự án nên người kiểm thử không thể tiến hành kiểm thử hết toàn bộ các giá trị đầu vào (Input). Lúc này, việc xác định tập các ca

kiểm thử đặc trưng sẽ xây dựng sao cho có thể bao phủ được tối đa các trường hợp là điều vô cùng cần thiết.

1.3.1. Kỹ thuật phân vùng tương đương

Kỹ thuật phân vùng tương đương có đặc điểm là:

- Chia miền dữ liệu đầu vào của một chương trình thành các vùng dữ liệu tương đương nhau.
- Tất cả các giá trị trong một vùng tương đương sẽ cho ra kết quả đầu ra giống nhau.
- Có thể chọn ra một giá trị đại diện trong một vùng tương đương để tiến hành kiểm thử.



Hình 8: Minh họa kỹ thuật phân vùng tương đương.

Việc thiết kế ca kiểm thử bằng kỹ thuật phân lớp tương đương dựa trên nguyên tắc xác định số vùng tương đương hợp lệ và số vùng tương đương không hợp lệ.

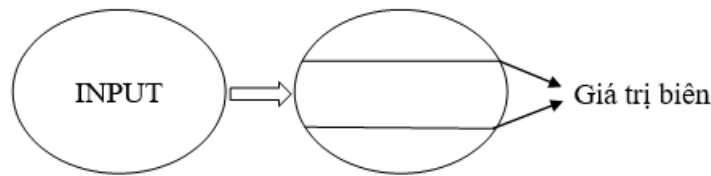
Ví dụ, trường hợp kiểm thử một ô textbox chỉ cho phép nhập vào số ký tự trong khoảng [5 - 30]. Áp dụng nguyên tắc xác định số vùng tương đương ta sẽ có các ca kiểm thử sau:

- Nhập vào một giá trị trong vùng tương đương không hợp lệ thứ nhất: Nhập 4 ký tự.
- Nhập vào một giá trị trong vùng tương đương hợp lệ: Nhập 6 ký tự.
- Nhập vào một giá trị trong vùng tương đương không hợp lệ thứ hai: Nhập 31 ký tự.

Như vậy với kỹ thuật trên, kiểm thử viên đã rút ngắn được số ca kiểm thử cần sinh ra so với việc phải kiểm thử toàn bộ các giá trị đầu vào.

1.3.2. Kỹ thuật phân tích giá trị biên

Phân tích giá trị biên tập trung vào các giá trị tại biên của miền xác định để xây dựng ca kiểm thử. Mục đích là tìm ra lỗi có thể xảy ra ở gần các giá trị biên này. Phân tích giá trị biên chính là trường hợp đặc biệt của kỹ thuật phân vùng tương đương. Dựa trên những vùng giá trị tương đương, kiểm thử viên sẽ xác định giá trị biên giữa những vùng này và thiết kế ca kiểm thử phù hợp.



Hình 9: Minh họa kỹ thuật phân tích giá trị biên.

Với kỹ thuật phân tích giá trị biên, kiểm thử viên cần chú ý tới một số giá trị sau để sinh ca kiểm thử:

- Giá trị nhỏ nhất.
- Giá trị gần kề lớn hơn giá trị nhỏ nhất.
- Giá trị gần kề nhỏ hơn giá trị nhỏ nhất.
- Giá trị bình thường.
- Giá trị gần kề nhỏ hơn giá trị lớn nhất
- Giá trị lớn nhất.
- Giá trị gần kề lớn hơn giá trị lớn nhất

Ví dụ: Kiểm thử một textbox nhập tuổi cho phép nhập giá trị số trong khoảng [0 - 150]. Vậy có thể sinh ra các ca kiểm thử cho trường hợp này theo kỹ thuật phân tích giá trị biên như sau:

- Giá trị nhỏ nhất: 0
- Giá trị gần kề lớn hơn giá trị nhỏ nhất: 1
- Giá trị gần kề lớn hơn giá trị nhỏ nhất: -1
- Giá trị bình thường: 70
- Giá trị gần kề nhỏ hơn giá trị lớn nhất: 149
- Giá trị lớn nhất: 150
- Giá trị gần kề lớn hơn giá trị lớn nhất: 151

Như vậy, có thể thấy phân tích giá trị biên là kỹ thuật bổ sung cho kỹ thuật phân vùng tương đương, giúp kiểm thử viên sinh ca kiểm thử để kiểm tra các giá trị tại biên.

1.3.3. Đoán lỗi

Một kỹ thuật thiết kế ca kiểm thử khác là đoán lỗi. Kiểm thử viên phỏng đoán lỗi dựa trên trực giác và kinh nghiệm của mình, từ đó liệt kê các trường hợp có thể xảy ra lỗi và sinh ca kiểm thử. Khó có thể đưa ra một quy trình cho kỹ thuật kiểm thử đoán lỗi vì nó có tính trực giác cao và không thể dự đoán trước.

Trong một số trường hợp, kiểm thử viên có thể kết hợp với lập trình viên để tìm ra những trường hợp có thể bị bỏ sót trong quá trình viết đặc tả yêu cầu phần mềm và lập trình.

1.3.4. Kỹ thuật chuyển trạng thái

Kỹ thuật này dựa trên việc quan sát, theo dõi quá trình chuyển từ trạng thái này sang trạng thái khác khi có một hành động xảy ra trong chương trình phần mềm để phát hiện các lỗi có thể xảy ra mà các kỹ thuật trên có thể bỏ sót. Ví dụ điển hình cho kỹ thuật chuyển trạng thái là việc kiểm thử chức năng giỏ hàng trong các trang Web thương mại điện tử. Lỗi có thể xuất hiện mỗi khi thêm sản phẩm vào giỏ hàng, xóa sản phẩm khỏi giỏ hàng hay khi thanh toán các sản phẩm trong giỏ hàng đó. Công việc của kiểm thử viên là xem xét các điều kiện trạng thái, theo dõi quá trình chuyển đổi giữa các trạng thái, điều kiện nhập đầu vào và các sự kiện kích hoạt thay đổi trạng thái.

II. KIỂM THỬ ỨNG DỤNG TRÊN NỀN WEB BẰNG CÔNG CỤ SELENIUM

2.1. Công cụ kiểm thử tự động Selenium

2.1.1. Giới thiệu chung về Selenium

Selenium là một phần mềm mã nguồn mở - một công cụ kiểm thử phần mềm tự động để kiểm thử các ứng dụng trên nền Web. Năm 2004, Selenium được phát triển bởi ThoughtWorks với cái tên ban đầu JavaScriptTestRunner. Đến năm 2007, tác giả Jason Huggins rời ThoughtWorks và gia nhập Selenium Team (thuộc Google), từ đó tiếp tục phát triển Selenium như hiện nay.

Selenium không chỉ là một công cụ duy nhất mà là một bộ các công cụ giúp kiểm thử tự động các ứng dụng trên nền Web hiệu quả hơn, bao gồm 4 phần: Selenium IDE, Selenium RC, Selenium Grid, Selenium WebDriver.

Selenium IDE là một công cụ cho phép chúng ta ghi lại một kịch bản và tái sử dụng kịch bản đó. Nó hoạt động như một Add-on của trình duyệt Mozilla Firefox với giao diện trực quan, dễ sử dụng ngay cả với những kiểm thử viên không biết về code. Với Selenium IDE, chúng ta chỉ có thể ghi lại kịch bản (Record) trên trình duyệt Mozilla Firefox nhưng có thể tái sử dụng kịch bản này trên nhiều trình duyệt khác như Internet Explorer, Google Chrome, v.v.

Selenium RC cho phép các nhà phát triển tự động hóa quá trình kiểm thử bằng cách sử dụng bất kỳ ngôn ngữ lập trình nào, phát huy tối đa thế mạnh của Selenium trong kiểm thử đơn vị. Để dễ dàng hơn cho việc kiểm thử, Selenium RC cung cấp các API và thư viện cho mỗi ngôn ngữ được hỗ trợ: HTML, Java, Perl, PHP, Ruby, Python, C#.

Selenium WebDriver là phiên bản kế nhiệm của Selenium RC. Cũng giống như Selenium RC, Selenium WebDriver hỗ trợ viết kịch bản kiểm thử bằng các ngôn ngữ khác nhau như Java, .NET, PHP, Python, Perl, Ruby và kiểm thử viên có thể sử dụng các điều kiện if, else hay các vòng lặp để tăng tính chính xác cho kịch bản kiểm thử. Selenium WebDriver có kiến trúc khá đơn giản, điều khiển trình duyệt trực tiếp từ hệ điều hành.

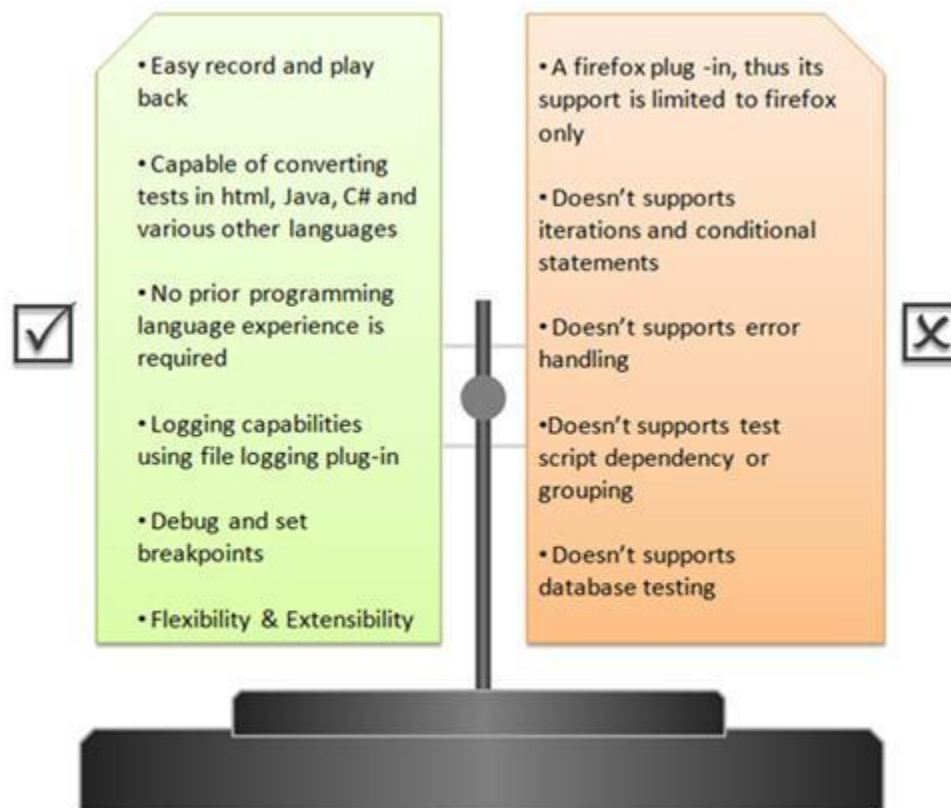
Selenium Grid là một hệ thống hỗ trợ kiểm thử viên thực thi kịch bản kiểm thử trên nhiều máy, nhiều trình duyệt một cách song song mà không cần chỉnh sửa kịch bản kiểm

thử. Ban đầu, Selenium Grid chỉ hỗ trợ cho Selenium RC nhưng sau này đã xuất hiện trên cả Selenium WebDriver. Selenium Grid cho phép kiểm thử viên thực thi ca kiểm thử trên nhiều máy khác nhau với nhiều trình duyệt khác nhau. Đặc biệt hơn, Selenium Grid còn cung cấp khả năng kiểm thử với chế độ phân tán.

2.1.2. Selenium IDE (Selenium Integrated Development Environment)

Selenium IDE được phát triển bởi Shinya Kasatani. Trong khi nghiên cứu về Selenium Core, ông đã nhận ra rằng code của JavaScript có thể mở rộng để tạo ra một môi trường phát triển tích hợp - integrated development environment (IDE). Với nó, người dùng có thể tích hợp vào trình duyệt Mozilla Firefox. IDE có khả năng ghi và chạy lại các hành động của người dùng trên trình duyệt Firefox - đã được tích hợp IDE. Sau này, Selenium IDE trở thành một phần của gói Selenium vào năm 2006. IDE có giá trị và tiềm năng lớn cho cộng đồng.

Selenium IDE là tool đơn giản và dễ sử dụng nhất trong tất cả các tool của gói Selenium. Tính năng record và play back làm nó đặc biệt đơn giản để học đối với những người đã biết chút về bất kỳ ngôn ngữ lập trình nào. Bên cạnh những điểm ưu việt nổi bật, một vài nhược điểm đi kèm của Selenium IDE làm nó không thích hợp để sử dụng trong trường hợp các kịch bản test thường thay đổi hoặc nâng cấp.



Advantages and disadvantages of Selenium IDE - © www.SoftwareTestingHelp.com

Hình 10: Ưu nhược điểm của Selenium IDE.

Ưu và nhược điểm của Selenium IDE

Ưu điểm	Nhược điểm
Dễ dàng ghi và chạy lại các thao tác	Chỉ có thể tích hợp trên trình duyệt Firefox
Có khả năng chuyển đổi test với html, C#, Java và các ngôn ngữ khác	Không hỗ trợ lặp lại và các câu lệnh có điều kiện
Không cần kinh nghiệm lập trình	Không hỗ trợ sửa lỗi
Có khả năng ghi log bằng việc sử dụng file logging plug-in trên Firefox	Không hỗ trợ test các kịch bản độc lập hoặc theo nhóm
Có thể debug và set breakpoints	Không hỗ trợ test database
Linh hoạt và có khả năng mở rộng	

Nhược điểm của IDE không thực sự là nhược điểm của Selenium. Hơn thế, đó chỉ là những hạn chế của IDE. Các hạn chế này có thể được khắc phục bằng cách sử dụng Selenium RC hoặc WebDriver.

2.1.3. Selenium RC (Selenium Remote Control)

Selenium RC là 1 tool được viết bằng Java, cho phép người dùng xây dựng kịch bản tes cho các ứng dụng dạng web với các ngôn ngữ lập trình mà họ chọn. Selenium RC là kết quả sau khi khắc phục được các hạn chế của Selenium IDE/ Core.

Lỗi hỏng và hạn chế là những thứ luôn tồn tại khi sử dụng Selenium Core, nó gây khó khăn cho người dùng để tận dụng hết lợi ích của tool này. Do đó, nó làm cho quá trình test trở thành nhiệm vụ công kênh và có ảnh hưởng sâu rộng.

Một trong những hạn chế chủ yếu là Chính sách Same Origin.

Các vấn đề với Same Origin Policy:

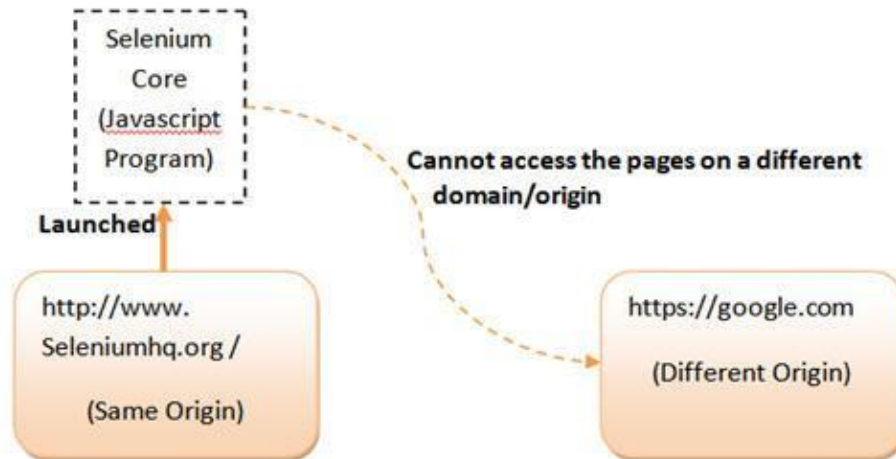
- Đó là không cho phép truy cập DOM của một tài liệu từ một nguồn khác với nguồn mà chúng ta đang cố truy cập tài liệu.
- Origin là một sự kết hợp tuần tự của lược đồ, máy chủ, cổng của Url.

Do đó, Selenium Core không thể truy cập các phần tử - elements từ một nguồn khác với nguồn mà nó được khởi chạy.

Để test với bất kỳ ứng dụng nào mà sử dụng Selenium Core, người ta phải cài đặt toàn bộ ứng dụng đó trên Selenium Core như một server để khắc phục các vấn đề về chính sách Same origin.

Selenium RC là 1 tool được viết bằng Java, cho phép người dùng xây dựng kịch bản tes cho các ứng dụng dạng web với các ngôn ngữ lập trình mà họ chọn. Selenium RC là kết quả sau khi khắc phục được các hạn chế của Selenium IDE/ Core.

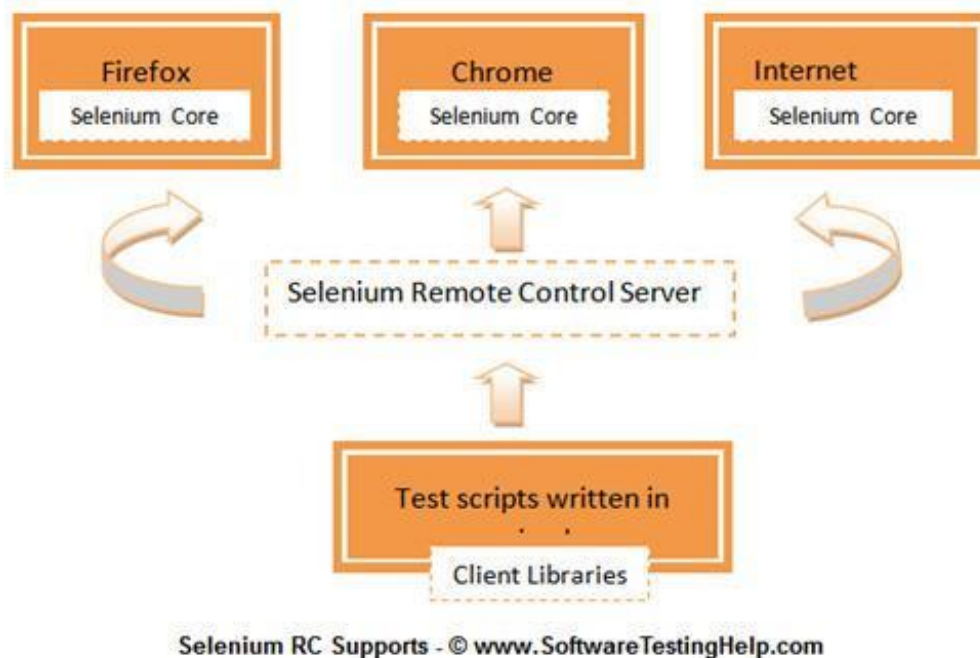
Lỗi hỏng và hạn chế là những thứ luôn tồn tại khi sử dụng Selenium Core, nó gây khó khăn cho người dùng để tận dụng hết lợi ích của tool này. Do đó, nó làm cho quá trình test trở thành nhiệm vụ cồng kềnh và có ảnh hưởng sâu rộng.



Hình 11: Minh họa test ứng dụng nào mà sử dụng Selenium Core.

Vì vậy, để điều chỉnh chính sách Same origin mà không cần tạo một bản sao chép ứng dụng để test trên Selenium Core, Selenium Remote Control được đề xuất. Trong khi Jason Huggins đang demo selenium, một đồng nghiệp khác ở ThoughtWorks tên là Paul Hammant đã gợi ý một cách giải quyết cho chính sách Same Origin và một công cụ có thể được kết nối với một ngôn ngữ lập trình mà họ đã chọn. Do đó, Selenium RC ra đời.

Không giống Selenium IDE, Selenium RC hỗ trợ nhiều trình duyệt và nền tảng.



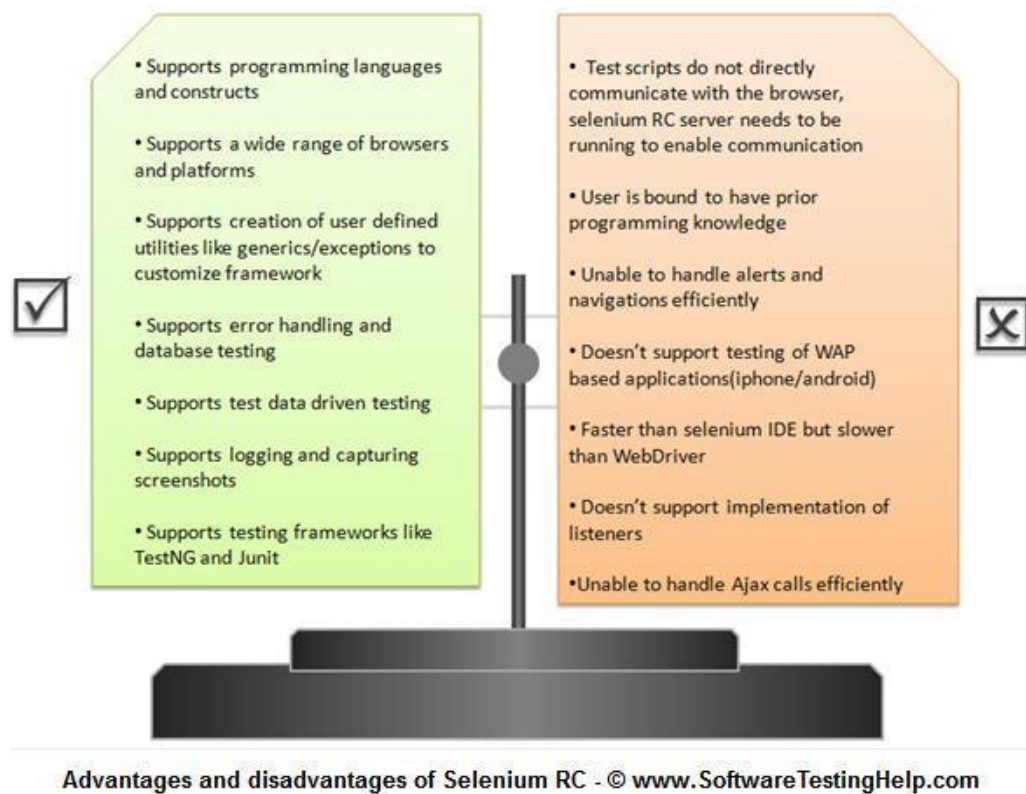
Hình 12: Hỗ trợ từ Selenium RC.

Mô tả luồng thực hiện

- Người dùng tạo ra kịch bản test bằng ngôn ngữ lập trình mong muốn.
- Với từng ngôn ngữ lập trình, sẽ có thư viện client riêng.
- Thư viện này sẽ xuất các câu lệnh test tới server selenium.
- Server selenium giải mã và chuyển đổi câu lệnh test sang lệnh JavaScript và gửi tới trình duyệt.
- Trình duyệt thực thi các câu lệnh bằng Selenium Core và gửi kết quả trở lại server Selenium.
- Server Selenium chuyển kết quả test cho thư viện client.

Các điều kiện tiên quyết cần thiết đặt trước khi tạo kịch bản Selenium RC:

- Ngôn ngữ lập trình: Java, C#, Python...



Hình 13: Ưu nhược điểm của Selenium RC.

- Môi trường phát triển tích hợp: Eclipse, Netbeans...
- Framework testing (tùy chọn): JUnit, TestNG...
- Selenium RC: setup off.

Ưu và nhược điểm của Selenium RC:

Ưu điểm	Nhược điểm
Hỗ trợ ngôn ngữ lập trình và cấu trúc	Kịch bản test không tương tác trực tiếp với trình duyệt, server Selenium RC cần được chạy để có thể tương tác.

Ưu điểm	Nhược điểm
Hỗ trợ đa trình duyệt và đa nền tảng	Người sử dụng buộc phải biết ngôn ngữ lập trình
Hỗ trợ tạo các tiện ích người dùng như Generics/ Exceptions để tùy chỉnh Framework	Không thể xử lý cảnh báo và điều hướng hiệu quả
Hỗ trợ xử lý lỗi và kiểm tra Database	Không hỗ trợ test các ứng dụng có nền tảng WAP (iphone/android)
Hỗ trợ test Data driven testing	Nhanh hơn selenium IDE nhưng chậm hơn WebDriver
Hỗ trợ ghi log và chụp màn hình	Không hỗ trợ triển khai
Hỗ trợ Framework testing giống như TestNG và JUnit	Không thể xử lý tốt gọi Ajax

2.1.4. Selenium WebDriver

Selenium WebDriver được tạo bởi kỹ sư khác tại ThoughtWorks tên là Simon Stewart vào năm 2006. WebDriver thì cũng là tool test web nhưng có sự khác biệt với Selenium RC. Ở thời điểm tool được xây dựng ở mức cơ bản, client cô lập đã được tạo ra cho mỗi trình duyệt; không cần nâng cấp JavaScript Heavy. Điều này dẫn tới việc phân tích sự tương thích giữa Selenium RC và WebDriver. Do đó, một tool tự động mạnh mẽ ra đời - được gọi là Selenium2.

WebDriver rõ ràng và là một framework hướng đối tượng. Nó sử dụng tính tương thích của các trình duyệt để tự động hóa mà không sử dụng bất cứ các thực thể ngoại vi nào. Với các nhu cầu ngày càng cao, nó trở nên phổ biến và có nhiều người sử dụng.

Ưu và nhược điểm của Selenium WebDriver:

Ưu điểm	Nhược điểm
Không yêu cầu có server selenium để chạy kịch bản	API phức tạp.
Tương tác trực tiếp với trình duyệt	Người sử dụng cần có kiến thức về lập trình
Giao diện hướng đối tượng	Không hỗ trợ test mobile
Hỗ trợ công cụ tìm kiếm động	Việc migrate từ selenium RC sang WebDriver là một quá trình mệt mỏi
Cung cấp tiện ích và class hỗ trợ trong việc xử lý cảnh báo, điều hướng, gọi Ajax và dropdowns	Không thể test ứng dụng có sử dụng đối tượng flash/flex
Hỗ trợ test các ứng dụng WAP (iphone/android)	

Hỗ trợ thực thi của người dùng	
Nhanh hơn Selenium RC	

2.1.5. Selenium Grid

Với Selenium RC, Tester cảm thấy tích cực và thuận lợi cho đến khi các xu hướng mới được đưa ra, yêu cầu thực hiện các kịch bản giống hoặc khác nhau trên nhiều nền tảng và trình duyệt, đồng thời để đạt được thực hiện kiểm thử phân tán, kiểm thử ở nhiều môi trường khác nhau và tiết kiệm đáng kể thời gian thực hiện. Vì vậy, Selenium Grid được đưa vào nhằm phục vụ các mục đích trên.

Selenium Grid được giới thiệu bởi Pat Lightbody để giải quyết các nhu cầu cho việc thực hiện các bộ test trên nhiều nền tảng cùng một lúc.

III. SỬ DỤNG SELENIUM IDE ĐỂ KIỂM THỬ NHỮNG CHỨC NĂNG CƠ BẢN WEBSITE SHOPEE

3.1. Các module kiểm thử

- Module 1: Kiểm tra phản hồi khi đăng nhập tài khoản
- Module 2: Kiểm tra phản hồi khi quên mật khẩu
- Module 3: Kiểm tra phản hồi khi tìm kiếm sản phẩm
- Module 4: Kiểm tra phản hồi khi mua hàng
- Module 5: Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng
- Module 6: Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng
- Module 7: Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm
- Module 8: Kiểm tra phản hồi khi đăng ký tài khoản mới

3.2. Xây dựng các module kiểm thử

3.2.1. Module 1 - Kiểm tra phản hồi khi đăng nhập tài khoản

Các test case có trong module 1:

Module Code		Module1							
Test requirement		Kiểm tra phản hồi khi đăng nhập tài khoản							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail	Untested	N/A					
7		0	0	0					
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module1 - 1]	Kiểm tra phản hồi khi đăng nhập tài khoản	Kiểm tra phản hồi khi nhập email và mật khẩu hợp lệ (mật khẩu đúng và đầy đủ các ký tự)	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhập mật khẩu của người dùng 4. Nhấn vào nút đăng nhập	Tài Khoản: locvnt02@gmail.com Mật khẩu: VuNguyenTienLoc3	Đăng nhập sẽ thành công	Đăng nhập thành công	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 2]		Kiểm tra phản hồi khi nhập email thiếu kí tự (test với kí tự '@')	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhấn vào nút đăng nhập	Tài Khoản: locvnt02@gmail.com	Đăng nhập thất bại	Đăng nhập thất bại (Tên tài khoản của bạn hoặc Mật khẩu không đúng, vui lòng thử lại)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 3]		Kiểm tra phản hồi khi nhập email chưa đăng ký tài khoản	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhấn vào nút đăng nhập	Tài Khoản: loc123456hihi@gmail.com	Đăng nhập thất bại	Đăng nhập thất bại (Tên tài khoản của bạn hoặc Mật khẩu không đúng, vui lòng thử lại)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 4]		Kiểm tra phản hồi khi nhập số điện thoại và mật khẩu hợp lệ (mật khẩu đúng và đầy đủ các ký tự)	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhập mật khẩu của người dùng 4. Nhấn vào nút đăng nhập	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3	Đăng nhập sẽ thành công	Đăng nhập thành công	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 5]		Kiểm tra phản hồi khi nhập số điện thoại thiếu số	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhấn vào nút đăng nhập	Tài Khoản: 098280106	Đăng nhập thất bại	Đăng nhập thất bại (Đăng nhập KHÔNG thành công. Bạn vui lòng thử lại hoặc đăng nhập bằng cách khác nhé!)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 6]		Kiểm tra phản hồi khi nhập số điện thoại chưa đăng ký tài khoản	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhấn vào nút đăng nhập	Tài Khoản: 0982801065	Đăng nhập thất bại	Đăng nhập thất bại (Tên tài khoản của bạn hoặc Mật khẩu không đúng, vui lòng thử lại)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module1 - 7]		Kiểm tra phản hồi khi nhập mật khẩu sai, thiếu kí tự	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Nhập tài khoản của người dùng 3. Nhập mật khẩu của người dùng 4. Nhấn vào nút đăng nhập	Tài Khoản: 0982801068 Mật khẩu: VuNguyen	Đăng nhập thất bại	Đăng nhập thất bại (Tên tài khoản của bạn hoặc Mật khẩu không đúng, vui lòng thử lại)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 14: Module 1 - Kiểm tra phản hồi khi đăng nhập tài khoản.

3.2.2. Module 2 - Kiểm tra phản hồi khi quên mật khẩu

Các test case có trong module 2:

Module Code		Module 2							
Requirement		Kiểm tra phản hồi khi quên mật khẩu							
Tester		Vũ Nguyễn Tiến Lộc							
Status		Fail	Unexecuted	N/A					
Count		1	0	0					
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
Module 2 - 1]	Kiểm tra phản hồi khi quên mật khẩu	Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập đúng tài khoản và nhập đúng mã xác minh trong thời gian cho phép	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh: ***** (mã chính xác)	Thành công vào được trang đối mặt khẩu	Thành công vào được trang đối mặt khẩu	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 2]		Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập đúng tài khoản và nhập đúng mã xác minh ngoài thời gian cho phép	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh: ***** (mã chính xác)	Không vào được trang đối mặt khẩu	Thành công vào được trang đối mặt khẩu	Fail	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 3]		Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập sai mã xác minh trong thời gian cho phép	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh: (mã sai)	Mã xác minh không hợp lệ	Mã xác minh không hợp lệ	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 4]		Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập đúng tài khoản và nhập sai mã xác minh trong thời gian cho phép và nhập lại một lần nữa mã xác minh sai khác	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh1: 000000 (mã sai) Mã xác minh2: 111111 (mã sai)	Mã xác minh không hợp lệ	Mã xác minh không hợp lệ	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 5]		Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập đúng tài khoản và nhập sai mã xác minh trong thời gian cho phép; sau đó nhập lại nhiều lần nữa mã xác minh sai khác (đến khi hết thời gian cho phép)	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh1: 000000 (mã sai) Mã xác minh2: 111111 (mã sai) ----- Mã xác minh3: 999999 (mã sai)	Mã xác minh không hợp lệ	Mã xác minh không hợp lệ	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 6]		Kiểm tra phản hồi khi nhấn quên mật khẩu, nhập đúng tài khoản và nhập sai mã xác minh trong thời gian cho phép; sau đó nhập đúng mã xác minh trong thời gian cho phép	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh	Tài khoản: 0982801060 Mã xác minh1: 111111 (mã sai) Mã xác minh2: ***** (mã chính xác)	Thành công vào được trang đối mặt khẩu	Thành công vào được trang đối mặt khẩu	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
Module 2 - 7]		Kiểm tra phản hồi khi gửi mã xác minh quên mật khẩu quá 10 lần	1. Mở trang web theo đường link: https://ahopos.vn/buyar/login?max=0&https%3A%2F%2Fahopos.vn%2F 2. Nhấn quên mật khẩu 3. Nhập tài khoản của người dùng 4. Nhấn vào nút tiếp tục 5. Chọn phương thức gửi mã xác minh 6. Chờ mã xác minh được gửi đến 7. Nhập mã xác minh 8. Nhấn tiếp tục	Tài khoản: 0982801060 Mã xác minh1: 000000 (mã sai) Mã xác minh2: 111111 (mã sai) ----- Mã xác minh3: 999999 (mã sai)	Không thể gửi mã xác minh một được nữa	Tài khoản không được gửi mã xác minh nữa (do vượt quá số lần cho phép gửi mã. Vui lòng thử lại sau 24h)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 15: Module 2 - Kiểm tra phản hồi khi quên mật khẩu.

3.2.3. Module 3 - Kiểm tra phản hồi khi tìm kiếm sản phẩm

Các test case có trong module 3:

Module Code		Module3							
Test requirement		Kiểm tra phản hồi khi tìm kiếm sản phẩm							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail	Untested		N/A				
4		2	0		0				

ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module3 - 1]	Kiểm tra phản hồi khi tìm kiếm sản phẩm	Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng việt có dấu, có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuot máy tính	Thành công hiện lên các sản phẩm theo từ khoá 'Chuot máy tính'	Đã thành công hiện lên các sản phẩm theo từ khoá 'Chuot máy tính'	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module3 - 2]		Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng việt có dấu, không có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuotmáy tính	Thành công hiện lên các sản phẩm theo từ khoá 'Chuotmáy tính' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Hiện lên các sản phẩm theo từ khoá 'Chuotmáy tính' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module3 - 3]		Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng việt không dấu, có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuot may tính	Thành công hiện lên các sản phẩm theo từ khoá 'Chuot may tính' (vẫn hiện lên giống như với 'Chuot máy tính')	Đã thành công hiện lên các sản phẩm theo từ khoá 'Chuot may tính' (vẫn hiện lên giống như với 'Chuot máy tính')	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module3 - 4]		Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng việt không dấu, không có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuotmay tính	Thành công hiện lên các sản phẩm theo từ khoá 'Chuotmay tính' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Đã hiện lên các sản phẩm theo từ khoá 'Chuotmay tính' khác với sản phẩm theo từ khoá 'Chuot máy tính'	Fail	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module3 - 5]		Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng anh, có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Computer mouse	Thành công hiện lên các sản phẩm theo từ khoá 'Computer mouse' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Đã hiện lên các sản phẩm theo từ khoá 'Computer mouse' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module3 - 6]		Kiểm tra phản hồi khi tìm kiếm với từ khoá tiếng anh, không có dấu cách	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Computermouse	Thành công hiện lên các sản phẩm theo từ khoá 'Computer mouse' giống với sản phẩm theo từ khoá 'Chuot máy tính'	Đã hiện lên các sản phẩm theo từ khoá 'Computer mouse' khác với sản phẩm theo từ khoá 'Chuot máy tính'	Fail	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 16: Module 3 - Kiểm tra phản hồi khi tìm kiếm sản phẩm.

3.2.4. Module 4 - Kiểm tra phản hồi khi mua hàng

Các test case có trong module 4:

Module Code		Module4							
Test requirement		Kiểm tra phản hồi khi mua hàng							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail	Untested		N/A				
3		0	0		0				
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module4 - 1]	Kiểm tra phản hồi khi mua hàng	Kiểm tra phản hồi mua hàng sau khi đăng nhập thành công	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm 6. Chọn sản phẩm muốn mua 7. Tiến hành thanh toán	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuột máy tính	Có thể mua được sản phẩm	Xác nhận thông tin thanh toán	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module4 - 2]		Kiểm tra phản hồi mua hàng sau khi đăng nhập thành công (nhưng không chọn mặt hàng nào)	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm 6. Chọn sản phẩm muốn mua 7. Tiến hành thanh toán	Tài Khoản: 0982801068 Mật khẩu: VuNguyenTienLoc3 Từ khoá: Chuột máy tính	Không thể thanh toán	Không thể thanh toán (Bạn vẫn chưa chọn sản phẩm nào để mua.)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module4 - 3]		Kiểm tra phản hồi mua hàng khi chưa đăng nhập thành công	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào thanh tìm kiếm 3. Gõ từ khoá 4. Nhấn vào biểu tượng tìm kiếm 5. Chọn sản phẩm muốn mua 6. Tiến hành thanh toán	Từ khoá: Chuột máy tính	Không thể thanh toán, bạn cần phải đăng nhập tài khoản trước	Không thể thanh toán, bạn cần phải đăng nhập tài khoản trước	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 17: Module 4 - Kiểm tra phản hồi khi mua hàng.

3.2.5. Module 5 - Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng

Các test case có trong module 5:

Module Code		Module5							
Test requirement		Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail	Untested		N/A				
1		0	0		0				
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module5 - 1]	Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng	Kiểm tra phản hồi khi đánh giá 5 sao sau khi mua hàng và nhận được hàng	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm 6. Chọn sản phẩm muốn mua 7. Tiến hành thanh toán 8. Nhận hàng và vào website đánh giá 5 sao	Từ khoá: Chuột máy tính	Đánh giá 5 sao thành công	Đánh giá 5 sao thành công	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 18: Module 5 - Kiểm tra phản hồi khi đánh giá sau khi mua hàng và nhận được hàng.

3.2.6. Module 6 - Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng

Module Code		Module6							
Test requirement		Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail		Untested		N/A			
2		0		0		0			
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module6 - 1]	Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng	Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng sau khi đăng nhập thành công	1. Mở trang web theo đường link: https://shopee.vn/buyer/login?next=https%3A%2F%2Fshopee.vn%2F 2. Đăng nhập tài khoản 3. Nhấn vào thanh tìm kiếm 4. Gõ từ khoá 5. Nhấn vào biểu tượng tìm kiếm 6. Chọn sản phẩm muốn mua 7. Thêm vào giỏ hàng	Từ khoá: Chuột máy tính	Thêm vào giỏ hàng thành công	Thêm vào giỏ hàng thành công	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module6 - 2]	Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng khi chưa đăng nhập thành công	Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng khi chưa đăng nhập thành công	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào thanh tìm kiếm 3. Gõ từ khoá 4. Nhấn vào biểu tượng tìm kiếm 5. Chọn sản phẩm muốn mua 6. Thêm vào giỏ hàng	Từ khoá: Chuột máy tính	Yêu cầu đăng nhập vào tài khoản trước khi thêm vào giỏ hàng	Yêu cầu đăng nhập vào tài khoản trước khi thêm vào giỏ hàng	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 19: Module 6 - Kiểm tra phản hồi khi thêm sản phẩm vào giỏ hàng.

3.2.7. Module 7 - Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm

Module Code		Module7							
Test requirement		Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm							
Tester		Vũ Nguyễn Tiến Lộc							
Pass		Fail		Untested		N/A			
2		0		0		0			
ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module7 - 1]	Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm theo giá giảm dần	Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm theo giá giảm dần	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào thanh tìm kiếm 3. Gõ từ khoá 4. Nhấn vào biểu tượng tìm kiếm 5. Chọn sản sắp xếp theo giá tăng dần	Từ khoá: Chuột máy tính	Sản phẩm hiển thị theo giá tăng dần	Sản phẩm hiển thị theo giá tăng dần	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module7 - 2]	Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm theo giá tăng dần	Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm theo giá tăng dần	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào thanh tìm kiếm 3. Gõ từ khoá 4. Nhấn vào biểu tượng tìm kiếm 5. Chọn sản sắp xếp theo giá giảm dần	Từ khoá: Chuột máy tính	Sản phẩm hiển thị theo giá giảm dần	Sản phẩm hiển thị theo giá giảm dần	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 20: Module 7 - Kiểm tra phản hồi khi sắp xếp thứ tự xuất hiện sản phẩm.

3.2.8. Module 8 - Kiểm tra phản hồi khi đăng ký tài khoản mới

Module Code	Module8		
Test requirement	Kiểm tra phản hồi khi đăng ký tài khoản mới		
Tester	Vũ Nguyễn Tiến Lộc		
Pass	Fail	Untested	N/A
7	0	0	0

ID	Test Suite	Test Case	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail	Tester	Test Date
[Module8 - 1]	Kiểm tra phản hồi khi đăng ký tài khoản mới	Kiểm tra phản hồi khi đăng ký tài khoản mới bằng số điện thoại chưa đăng ký, nhập mã xác minh đúng	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập số điện thoại 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Số điện thoại: 0936614892 Mã xác minh: ***** (mã chính xác)	Đăng ký thành công tài khoản mới	Đăng ký thành công tài khoản mới	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module8 - 2]		Kiểm tra phản hồi khi đăng ký tài khoản mới bằng số điện thoại chưa đăng ký, nhập mã xác minh sai	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập số điện thoại 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Số điện thoại: 0936614892 Mã xác minh: ***** (mã sai)	Mã xác minh không hợp lệ, vui lòng nhập lại	Mã xác minh không hợp lệ, vui lòng nhập lại	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module8 - 3]		Kiểm tra phản hồi khi đăng ký tài khoản mới bằng số điện thoại chưa đăng ký, nhập mã xác minh sai quá 10 lần	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập số điện thoại 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Số điện thoại: 0936614892 Mã xác minh: ***** (mã sai) X10	Không thể gửi mã xác minh mới được nữa	Tài khoản không được gửi mã xác minh nữa (do vượt quá số lần cho phép gửi mã. Vui lòng thử lại sau 24h)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module8 - 4]		Kiểm tra phản hồi khi đăng ký tài khoản mới bằng email chưa đăng ký, nhập mã xác minh đúng	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập email 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Email: callmeloc0612@gmail.com Mã xác minh: ***** (mã chính xác)	Đăng ký thành công tài khoản mới	Đăng ký thành công tài khoản mới	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module8 - 5]		Kiểm tra phản hồi khi đăng ký tài khoản mới bằng email chưa đăng ký, nhập mã xác minh sai	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập email 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Email: callmeloc0612@gmail.com Mã xác minh: ***** (mã sai)	Mã xác minh không hợp lệ, vui lòng nhập lại	Mã xác minh không hợp lệ, vui lòng nhập lại	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024
[Module8 - 6]		Kiểm tra phản hồi khi đăng ký tài khoản mới bằng email chưa đăng ký, nhập mã xác minh sai quá 10 lần	1. Mở trang web theo đường link: https://shopee.vn 2. Nhấn vào đăng ký 3. Nhập email 4. Nhập mã xác minh gửi về 5. Tiến hành nhập thông tin tài khoản	Email: callmeloc0612@gmail.com Mã xác minh: ***** (mã sai) X10	Không thể gửi mã xác minh mới được nữa	Tài khoản không được gửi mã xác minh nữa (do vượt quá số lần cho phép gửi mã. Vui lòng thử lại sau 24h)	Pass	Vũ Nguyễn Tiến Lộc	22/03/2024

Hình 21: Module 8 - Kiểm tra phản hồi khi đăng ký tài khoản mới.

IV. TÀI LIỆU THAM KHẢO

- [1] Kiểm thử và đảm bảo chất lượng phần mềm – TS. Nguyễn Đăng Khoa
- [2] Kiểm thử và đảm bảo chất lượng phần mềm - Nguyễn Đức Mận, Thạc Bình Cường
- [3] <https://www.selenium.dev/selenium-ide/>
- [4] <https://www.youtube.com/watch?v=m4KpTvEz3vg>
- [5] <https://www.browserstack.com/guide/what-is-selenium-ide>