

MỞ ĐẦU

Trong thời buổi công nghệ 4,0 như hiện nay, các tổ chức cá nhân, doanh nghiệp ngày càng phát triển vượt trội. Để đáp ứng nhu cầu thực hiện nhanh chóng, hiệu quả, tiện lợi,... không thể nào không kể đến các phần mềm, trang web. Sự lựa chọn sử dụng website là một trong những sự lựa chọn phổ biến nhất.

Website là nơi có thể chia sẻ thông tin, truyền đạt thông tin một cách tiện lợi nhanh chóng nhất, không giới hạn người dùng chính vì điều đó website có thể làm được mọi công việc đáp ứng được nhu cầu của bạn.

Tuy nhiên, đi cùng với triển nhanh chóng và tiện lợi thì cũng có các trở ngại dẫn đến website không thể hoạt động, hoạt động không đúng với mục đích mong muốn. Do đó, cần phải kiểm thử đoán lỗi để đảm bảo chất lượng trang website.

Với sự hướng dẫn của giáo viên em xin thực hiện đề tài "Kiểm thử đoán lỗi" và thực hiện trên một trang website đã được xây dựng nhưng chưa hoàn thiện, Do hạn chế về mặt kiến thức cho nên không thể tránh khỏi sai sót trong quá trình làm báo cáo, rất mong được sự giúp đỡ và chỉ dạy của thầy.

MỤC LỤC

PHẦN 1. PHẦN MỞ ĐẦU	1
1. Lý do chọn đề tài	1
2. Mục tiêu nghiên cứu.....	1
PHẦN II. TỔNG QUAN VỀ KIỂM THỬ WEBSITE	2
1. Thế nào là kiểm thử Website?	2
2. Vai trò của kiểm thử Website.....	2
3. Các cấp độ kiểm thử.....	2
4. Kỹ thuật kiểm thử website.....	5
PHẦN III: CÔNG CỤ KIỂM THỬ TỰ ĐỘNG SELENIUM	8
1. Khái quát về Selenium	8
2. Selenium IDE	9
PHẦN IV: MỘT SỐ LỖI.....	12
1. Lỗi logic	12
2. Lỗi giao diện người dùng	12
3. Lỗi tính toán:	12
4. Lỗi bảo mật	13
5. Lỗi Tương Thích:	14
6. Lỗi Tích Hợp:	15
PHẦN V: KẾT LUẬN	17
1. Kết quả đạt được	17
2. Hạn chế.....	17
3. Hướng phát triển đề tài.....	17
TÀI LIỆU THAM KHẢO.....	18

PHẦN 1. PHẦN MỞ ĐẦU

1. Lý do chọn đề tài kiểm thử đoán lỗi

Website được xem là hướng giải quyết mới cho sự phát triển của ứng dụng Internet. Đa dạng, sử dụng được nhiều công nghệ khác nhau, giúp cho nhiều ứng dụng sử dụng cùng một ngôn ngữ chung, có thể trao đổi với nhau qua mạng Internet. Website không chỉ kết nối được một trang web mà kết nối được với tất cả các trang web khác nhau, Không chỉ dừng lại ở đó chúng có thể tạo thành một mạng lưới liên kết rộng lớn trên toàn thế giới. Do đó nó sẽ mang đến nhiều trở ngại đáng kể cho các nhà kiểm thử đoán lỗi.

Năng suất giảm đồng nghĩa với việc trang website xuất hiện sai sót trong quá trình vận hành. Các lỗi thường là: lỗi bảo mật, sai chức năng,...Vậy muốn biết chúng sai ở đâu ta cần tìm đến các nhà kiểm thử đoán lỗi.

Để đưa ra được các giải pháp kiểm thử các nhà nghiên cứu, các chuyên gia phải thực hiện kiểm nghiệm chức năng, độ tin cậy của trang web,... và những vấn đề liên quan đến người sử dụng.

Với định hướng mong muốn tiếp cận nhiều hơn, có cái nhìn rộng, hiểu biết chính xác về kiểm thử đoán lỗi, đó cũng chính là lý do để em chọn đề tài “kiểm thử đoán lỗi website” để làm đồ án cơ sở ngành cho môn học.

2. Mục tiêu nghiên cứu

- Hiểu rõ thành phần của, các vấn đề cơ bản liên quan đến website, những lỗi thường xảy ra trong quá trình hoạt động của website.
- Biết thêm nhiều phần mềm kiểm thử giúp cho việc nghiên cứu trở nên dễ dàng và nhanh chóng hơn.
- Giúp em mở mang thêm nhiều kiến thức mới, là một phần kiến thức bổ ích cho định hướng tương lai của em sau này.

PHẦN II. TỔNG QUAN VỀ KIỂM THỬ WEBSITE

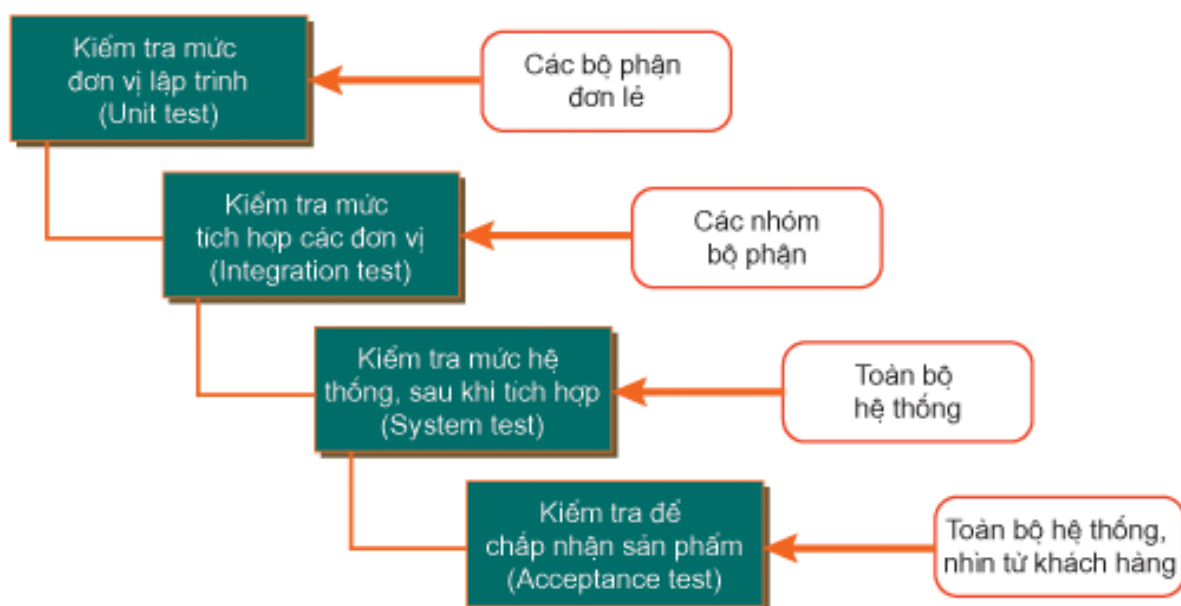
1. Thế nào là kiểm thử Website?

- Kiểm thử website là tên gọi được đặt cho một quá trình kiểm thử phần mềm tập trung vào việc kiểm tra các ứng dụng web. Ứng dụng web cần được kiểm tra hoàn toàn trước khi đi vào hoạt động, điều này có thể giúp giải quyết các vấn đề trong ứng dụng web trước khi tiếp xúc với người dùng như các vấn đề về chức năng, bảo mật, các vấn đề dịch vụ web, các vấn đề tích hợp và khả năng xử lý lưu lượng truy cập, trong quá trình kiểm thử website, cần cố gắng phát hiện ra lỗi có thể xảy ra trong hệ thống nhằm giải quyết kịp thời.
- Kiểm thử website chính là kiểm tra xem ứng dụng web có chứa những lỗi tiềm tàng nào không, trước khi chính thức đưa website đi vào sử dụng. Đây là một công việc liên quan đến lập trình web app, các khâu cần kiểm thử đó là bảo mật, chức năng, khả năng xử lý lưu lượng, hiệu suất trang web.
- Nhu cầu sử dụng internet của mọi người tăng cao, do đó việc thiết kế website (ứng dụng web) cũng phát triển rất mạnh mẽ. Website ngày càng đóng vai trò quan trọng trong thương mại điện tử, trao đổi thông tin, giải trí, thư giãn, giới thiệu,... Muốn thiết kế website có hiệu suất sử dụng cao, thì khi sau thiết kế xong, cần phải kiểm thử trang web đó thật cẩn thận, chặt chẽ.

2. Vai trò của kiểm thử Website

- Website đã đưa ra nhiều thách thức lớn cho nhà phát triển
- Chất lượng của một trang website sẽ phụ thuộc vào: thành phần trình chủ, trình khách, cấu trúc hệ thống và khả năng tương tác. Muốn một trang website có thể tạo ra sản phẩm tốt cần phải kiểm thử.
- Mục đích của kiểm thử nhằm tìm ra các lỗi thiếu sót trong một website. Tìm ra cách giải quyết nhanh nhất, hiệu quả nhất và chắc chắn rằng lỗi sẽ được sửa lại.

3. Các cấp độ kiểm thử



Cấp độ kiểm thử

3.1. Kiểm thử đơn vị

- Kiểm thử đơn vị còn được gọi là kiểm thử thành phần, phổ biến nhất là kiểm thử từng chức năng của mã. Kiểm thử đơn vị bao gồm: hàm, thủ tục, lớp và phương thức.
- Kiểm thử đơn vị sẽ do một lập trình viên thực hiện. Trong suốt giai đoạn viết code việc kiểm thử đơn vị cần thực hiện sớm, nhanh chóng.
- Việc kiểm thử đơn vị sẽ giúp bảo vệ thông tin được xử lý và xuất chính xác nhất, trong mối tương quan của dữ liệu nhập và xử lý. Điều này sẽ đòi hỏi các nhánh Unit phải kiểm tra và phát hiện nhánh có thể phát sinh ra các lỗi.

3.2. Kiểm thử tích hợp

- Kiểm thử tích hợp là hình thức kiểm thử kết hợp các thành phần của website xem như đó là một trang web đã được hoàn thiện. Kiểm thử tích hợp là để kiểm tra sự giao tiếp của các thành phần
- Phát hiện lỗi giao tiếp

- Tích hợp các thành phần đơn lẻ thành một hệ thống nhỏ và tạo thành một hệ thống hoàn chỉnh nhất.
- Kiểm thử tích hợp gồm 4 loại:
 - Kiểm thử cấu trúc: Kiểm thử để xem chương trình bên trong của trang website chạy đúng không, quan tâm đến các thành phần nội tại như: các lệnh, các nhánh ở bên trong.
 - Kiểm thử chức năng: quan tâm đến chức năng của chương trình, khảo sát chức năng chương trình theo yêu cầu kỹ thuật
 - Kiểm thử hiệu năng: Kiểm thử xem sự hoạt động của hệ thống
 - Kiểm thử khả năng chịu tải: Kiểm thử giới hạn chịu tải lớn nhất của một hệ thống website.

3.3. Kiểm thử hệ thống

- Kiểm thử hệ thống là kiểm thử xem toàn bộ hệ thống thiết kế đã đúng với mục đích đặt ra ban đầu hay chưa.
- Kiểm tra các hành vi chức năng, yêu cầu chất lượng, độ tin cậy, tiện lợi và khả năng bảo mật của website.
- Việc này được bắt đầu khi toàn bộ trang website đã được tích hợp hoàn toàn. Kiểm thử hệ thống đa phần sẽ mất rất nhiều thời gian lẫn công sức. Trong một số trường hợp kiểm thử đòi hỏi nhiều thiết bị để hỗ trợ cho phần mềm hoặc phần cứng. Đối với mức độ hệ thống thì người kiểm tra sẽ quan tâm đến việc đánh giá hoạt động, thao tác và tin cậy và các yêu cầu liên quan khác.
- Kiểm thử hệ thống gồm các loại như sau:
 - Kiểm thử chức năng: đảm bảo toàn bộ hệ thống hoạt động đúng với yêu cầu thiết kế
 - Kiểm thử hiệu năng: kiểm tra sự tuân thủ của hệ với các yêu cầu của hiệu năng, đảm bảo chất lượng hệ thống: khả năng mở rộng, độ tin cậy,...
 - Kiểm thử cơ sở dữ liệu: kiểm tra hiển thị của hệ thống có giống với cơ sở dữ liệu hiển thị dữ liệu hay không

- Kiểm thử khả năng bảo mật: Kiểm tra xem hệ thống đã an toàn hay chưa, đảm bảo dữ liệu không bị đánh cắp.
- Kiểm tra tính khả dụng: kiểm tra tính dễ sử dụng của hệ thống xem có thân thiện với người dùng hay không.
- Kiểm tra tính tương thích: kiểm tra hệ thống tương thích với các yếu tố với các yếu tố mà nó hoạt động không (trình duyệt, hệ điều hành,...)
- Kiểm tra khả năng phục hồi: Kiểm tra khả năng khôi phục trạng thái ổn định khi trang website gặp các sự cố xảy ra.

3.4. Kiểm thử chấp nhận

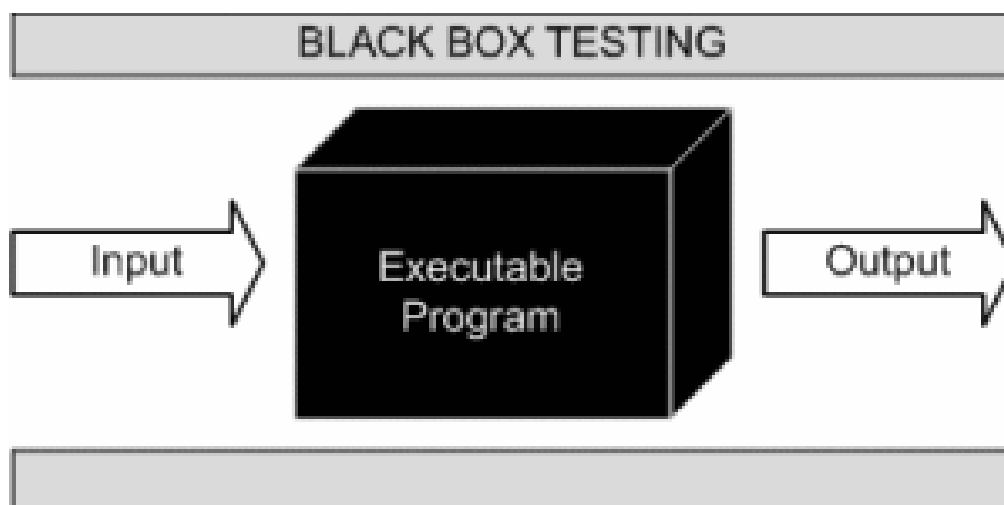
- Kiểm thử chấp nhận là kiểm thử chắc chắn rằng trang website đáp ứng được nhu cầu người dùng.
- Kiểm thử chấp nhận người kiểm tra sẽ là khách hàng. Khách hàng sẽ đánh giá trang website dựa theo khả năng sử dụng thường xuyên của họ. Kiểm tra ở giai đoạn tránh cho việc trang website hoạt động không đúng yêu cầu, không đáp ứng được nhu cầu người dùng.

4. Kỹ thuật kiểm thử website

Mục đích của việc kiểm thử là đặt ra các trường hợp phát hiện lỗi cao nhất và nhanh nhất với thời gian và công sức tối thiểu. Vì vậy nên kiểm thử website được chia thành hai loại:

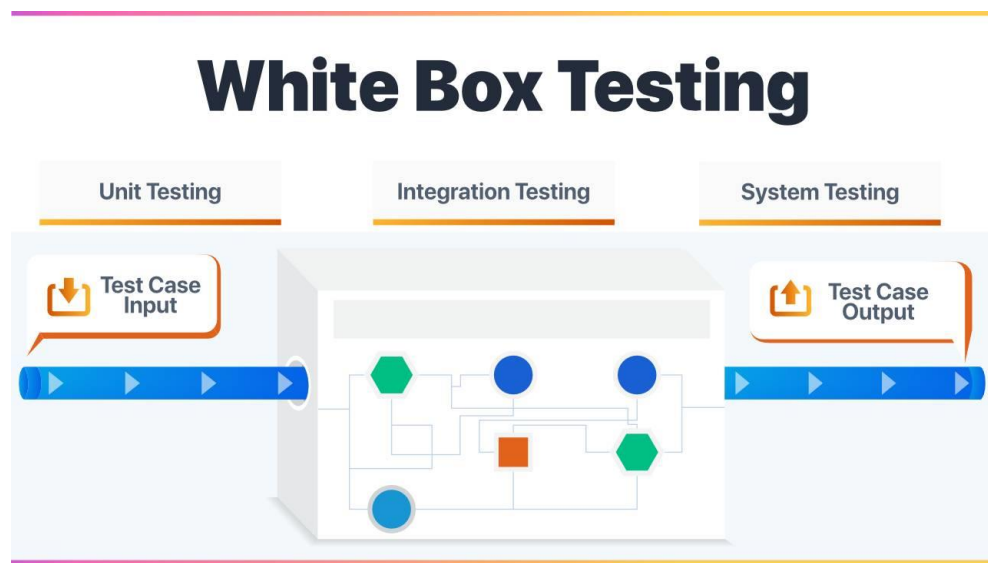
- Kỹ thuật kiểm thử hộp đen: còn có tên gọi khác là kiểm thử chức năng.
- Kỹ thuật kiểm thử hộp trắng: còn có tên gọi là kiểm thử cấu trúc.

4.1. Kỹ thuật kiểm thử hộp đen



- Kỹ thuật kiểm thử hộp đen là kỹ thuật kiểm thử hướng đến phần dữ liệu website hoặc có thể hiểu là kiểm thử đầu vào và kiểm thử đầu ra.
- Đối với kỹ thuật này khi kiểm thử, người ta sẽ xem website như một hộp đen, sẽ không chú trọng đến cấu trúc hoặc các hành vi của chương trình. Chỉ cần quan tâm đến các hướng mà trang website không hoạt động đúng với đặc tả ban đầu của nó.
- Kiểm thử hộp đen sẽ chú trọng đến chức năng. Người kiểm thử sẽ xây dựng một nhóm giá trị đầu vào thực thi đầy đủ các yêu cầu chức năng của chương trình. Phương pháp này sẽ cố gắng tìm ra các lỗi:
 - Chức năng bị thiếu hoặc không được chính xác
 - Lỗi giao diện
 - Lỗi cấu trúc dữ liệu truy cập
 - Lỗi hành vi, hiệu suất
 - Lỗi khởi tạo và kết thúc

4.2. Kỹ thuật kiểm thử hộp trắng



- Kiểm thử hộp trắng hay còn gọi là kiểm tra bên trong cấu trúc của trang website
- Mục đích của việc kiểm tra là nhằm tới cấu trúc của từng đoạn mã, vì khi một trang website được hình thành thì thành phần quan trọng của nó là các đoạn mã viết lên để tạo thành một cấu trúc của website. Do đó các lỗi cũng sẽ thường xuyên xảy ra ở đây. Kỹ thuật kiểm thử hộp trắng là kỹ thuật kiểm thử rất quan trọng trong kiểm thử một website.

PHẦN III: CÔNG CỤ KIỂM THỬ TỰ ĐỘNG SELENIUM

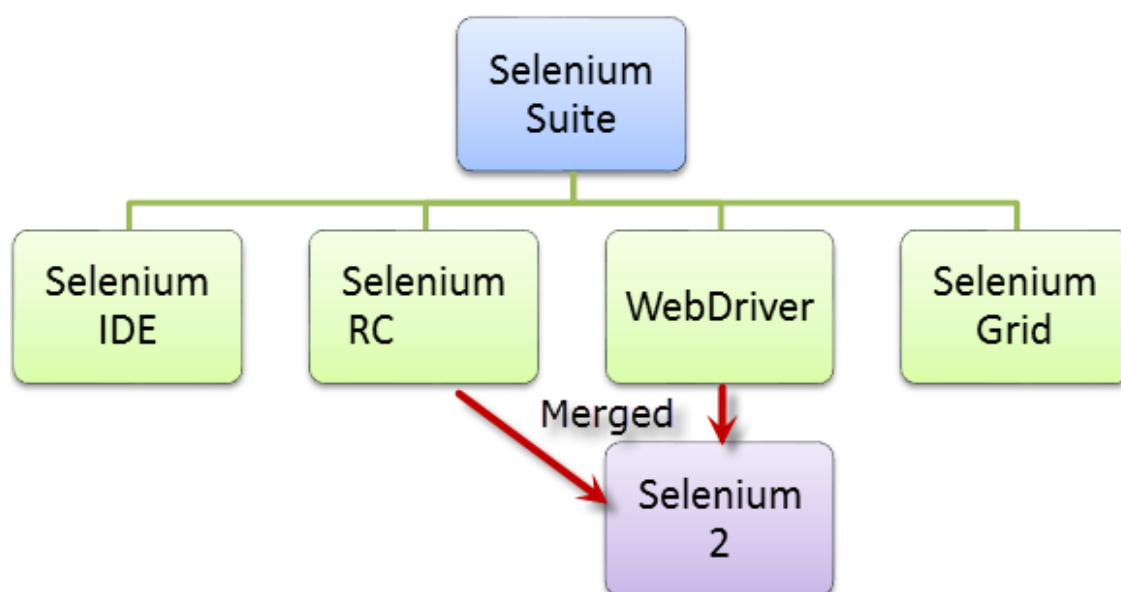
1. Khái quát về Selenium

1.1. Giới thiệu về Selenium?

- Selenium là công cụ thử nghiệm các ứng dụng web được phát triển vào năm 2004 bởi lập trình viên Jason Huggins làm việc tại Th ThinkWorks với tên gọi là JavaScript Test Runner. Ông đã phát triển thư viện Javascript để chạy tự động các test trên nhiều trình duyệt. Đây chính là cơ sở để Selenium IDE và Selenium RC ra đời.
- Năm 2006, Simon Stewart - một nhân viên của Google tiếp tục phát triển Selenium với công việc được đặt tên là WebDriver. Nhờ có công cụ này, Google đã nhận được một lượng người sử dụng Selenium rất lớn nhưng đứng trước những hạn chế của sản phẩm thì các tester vẫn phải làm việc rất vất vả.
- Năm 2008, Selenium và WebDriver chính thức được kết hợp bởi Selenium đang dần lớn mạnh và WebDriver lại là công cụ của tương lai.

1.2. Các thành phần của Selenium.

- Selenium gồm 4 thành phần chính: Selenium IDE, Selenium Remote Control (RC), Selenium Web Driver và Selenium Grid.



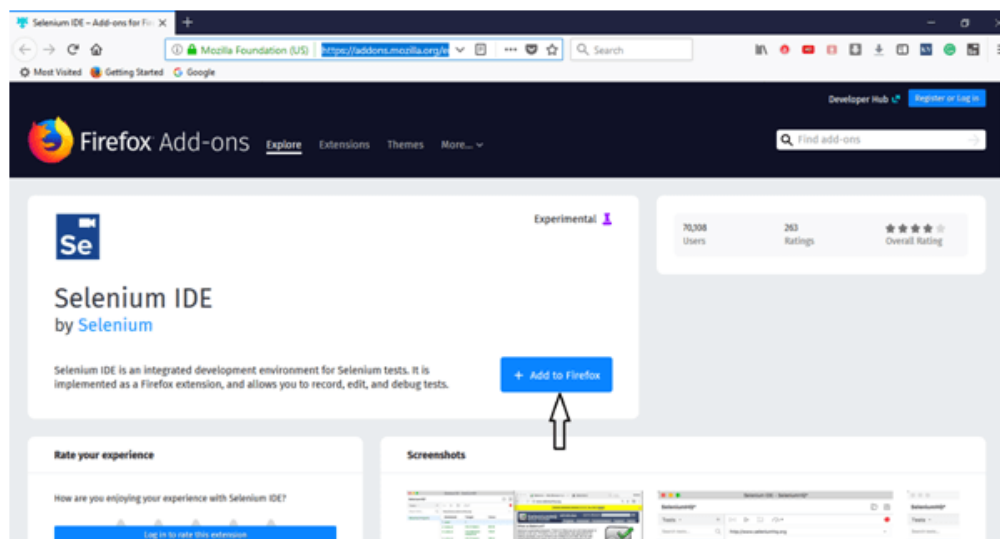
- Selenium IDE: xây dựng các trường hợp kiểm thử Selenium. Nó cung cấp một giao diện dễ sử dụng để chạy, phát triển các trường hợp kiểm thử. Selenium IDE có thể tái sử dụng kịch bản, tích hợp với trình duyệt Firefox. Tuy Selenium chỉ là một Firefox add-on dựa trên test case nhưng nó vẫn có thể chạy trên các trình duyệt nhờ vào sử dụng Selenium RC.
- Selenium Core: công cụ được tích hợp trong Selenium IDE. Đây là một công cụ được chạy bằng test script. Lợi thế của công cụ này có thể chạy trên các nền tảng trình duyệt khác nhau, tuy nhiên nó vẫn có nhược điểm là phải cài đặt máy chủ của website cần kiểm tra.
- Selenium RC: có thể tự động hóa kiểm thử và sử dụng một ngôn ngữ lập trình và tạo tính linh hoạt.
Selenium RC là thư viện cho các ngôn ngữ được hỗ trợ: HTML, Java, C#, Perl, PHP, Python, Ruby. Sử dụng Selenium với ngôn ngữ lập trình bậc cao để phát triển việc kiểm thử có thể thích nghi được với mọi dự án.
- Selenium Grid: là phương pháp kiểm tra phân bố, phối hợp với kết quả của Selenium IDE để thực hiện cùng một lúc trên nhiều trang website khác nhau và lưu lại kết quả.

2. Selenium IDE

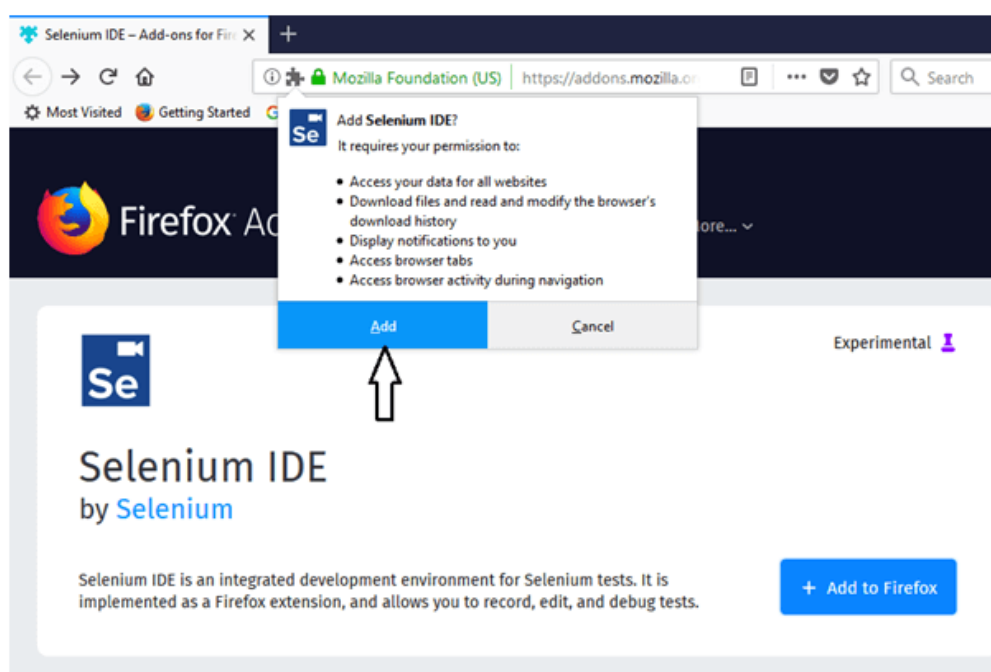
- Selenium IDE là add-on thuộc Mozilla Firefox phiên bản 2.0 trở nên, lúc đầu được phát triển bởi Shinya Kasatani theo hướng sử dụng Selenium Core không cần cài Selenium vào máy chủ.
- Cho phép ghi lại quá trình kiểm thử bằng các chức năng Record và Playback.

2.1. Cài đặt Selenium IDE

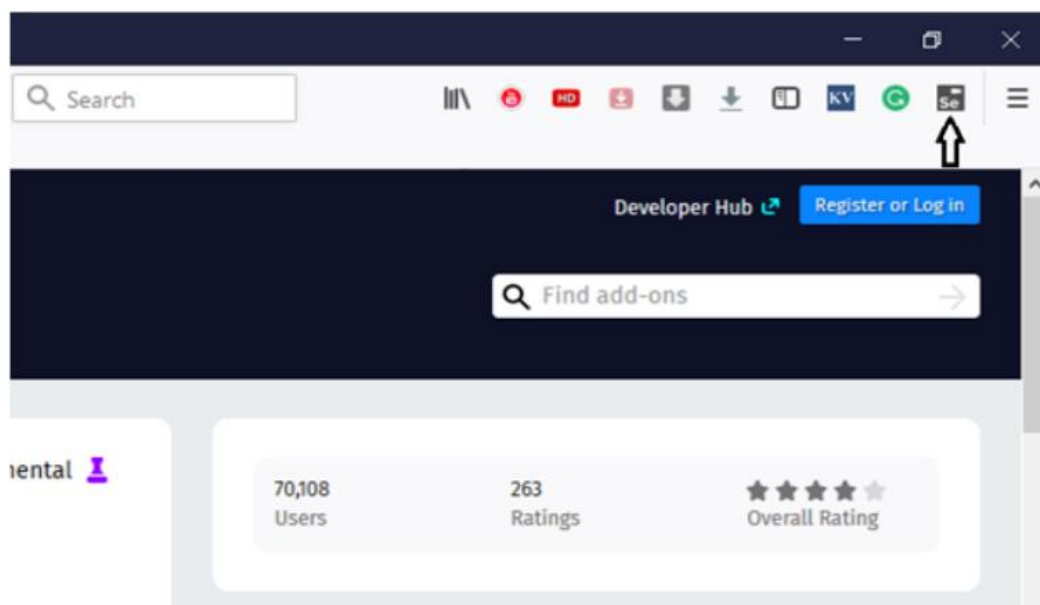
- Bước 1: Tìm trang
<https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/>
- Bước 2: Trang sẽ được chuyển đến tranh hỗ trợ chính thức của Fire Fox
- Bước 3: Nhấn nút “Add to Fire Fox”



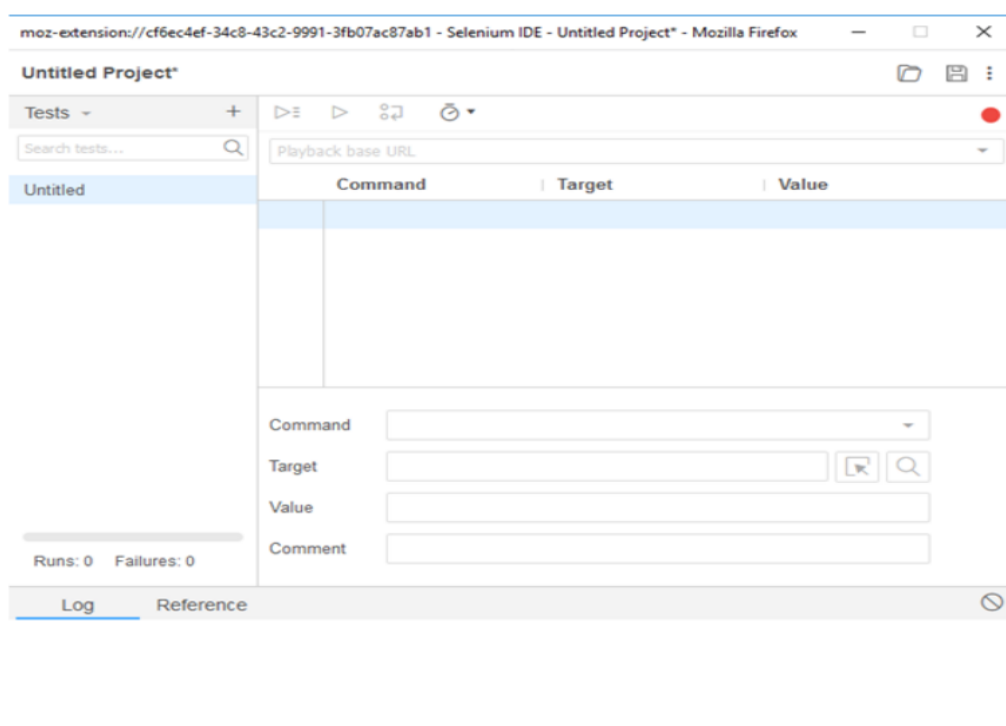
- Bước 4: Hộp thoại xuất hiện yêu cầu bạn thêm Selenium IDE làm tiện ích mở rộng cho trình duyệt đang sử dụng.
- Bước 6: Nhấn nút “thêm”



- Bước 7: Khởi động lại trình duyệt FireFox
- Bước 8: Đến góc trên cùng bên phải của trình duyệt và tìm biểu tượng Selenium IDE



- Bước 9: Nhấn vào biểu để khởi chạy Selenium IDE



PHẦN IV: MỘT SỐ LỖI

1. Lỗi logic

Lỗi logic là lỗi mà chương trình vẫn chạy mà không có bất kỳ thông báo lỗi nào nhưng nó không thực hiện đúng chức năng mà mình mong muốn. Lỗi này thường phát sinh do những sai sót trong việc xây dựng logic cho chương trình

Ví dụ như lỗi dưới đây lỗi trong (python):

```
if x > 0:
    print("Số dương")
elif x > 10:
    print("Số lớn hơn 10")
```

Trong đoạn mã trên, nếu x là một số lớn hơn 10, thì chỉ có thông báo "Số dương" sẽ được in trên màn hình, do điều kiện 'elif' không bao giờ được kiểm tra.

2. Lỗi giao diện người dùng

- Hiện thị chưa đúng: giao diện người dùng không hiển thị đúng thông tin, hình ảnh, hoặc điều kiện.
- Vấn đề về độ phân giải: Giao diện không hiển thị đúng trên các màn hình có độ phân giải khác nhau, dẫn đến mất thông tin hoặc trải nghiệm người dùng không tốt.
- Lỗi chuyển động: Nếu có các yếu tố chuyển động trong giao diện, lỗi có thể xảy ra khi chúng không mượt mà hoặc không hiển thị đúng cách.
- Chưa đáp ứng: Giao diện không phản hồi đúng cách khi người dùng tương tác, chẳng hạn như không có phản hồi khi nhấp vào các nút hoặc không di chuyển mượt mà.
- Lỗi không thể Đọc: Vấn đề với kiểu chữ, kích thước chữ, màu sắc chữ, hoặc nền làm cho nội dung trở nên khó đọc hoặc không thể đọc được.
- Lỗi phong chữ: Sự lựa chọn không đúng về phong chữ, kích thước chữ, hay khoảng cách giữa các chữ có thể gây khó khăn trong việc đọc và hiểu thông tin.
- Lỗi Vị trí: Các yếu tố trên giao diện không được đặt đúng vị trí, dẫn đến sự nhòe lạc hoặc không cân đối.

3. Lỗi tính toán:

- Lỗi Trong Công thức: Các lỗi có thể xuất hiện khi công thức tính toán không được viết đúng cách, dẫn đến kết quả không mong muốn.
- Lỗi Chia cho 0: Nếu có phép chia cho 0 trong quá trình tính toán, sẽ xuất hiện lỗi và ảnh hưởng đến tính đúng đắn của chương trình.
- Lỗi Tính toán số thập phân: Trong các hệ thống sử dụng số thực dấu động, có thể xảy ra sai số do cách máy tính lưu trữ và xử lý số thực.
- Lỗi Tính toán Số nguyên: Khi thực hiện phép toán trên số nguyên, có thể xảy ra tràn số (integer overflow) nếu giá trị tính toán vượt quá giới hạn của kiểu dữ liệu.
- Lỗi Tính toán Mảng: Khi làm việc với mảng, có thể xảy ra lỗi nếu không tính toán đúng với các chỉ số và giá trị của mảng.
- Lỗi Tính toán Dữ liệu Đầu vào: Nếu dữ liệu đầu vào không được xử lý đúng, các lỗi tính toán có thể xuất hiện và ảnh hưởng đến kết quả cuối cùng. Để ngăn chặn lỗi tính toán dữ liệu đầu vào, quan trọng để thực hiện kiểm thử đầu vào kỹ lưỡng, áp dụng các phương pháp kiểm thử biên và xác nhận rằng chương trình xử lý đúng mọi loại dữ liệu có thể xuất hiện. Các bước kiểm tra lỗi và xử lý đầu vào là quan trọng để đảm bảo tính độ tin cậy và đúng đắn của hệ thống.

4. Lỗi bảo mật

- Lỗi bảo mật là những vấn đề trong phần mềm mà có thể tạo điều kiện cho các mối đe dọa và tấn công từ bên ngoài. Việc giải quyết lỗi bảo mật là một phần quan trọng của quá trình phát triển phần mềm để đảm bảo rằng hệ thống được bảo vệ một cách tốt nhất. Dưới đây là một số loại lỗi bảo mật phổ biến:
- SQL: Đây là một loại tấn công trong đó kẻ tấn công chèn các lệnh SQL độc hại vào các truy vấn SQL, có thể làm lộ thông tin nhạy cảm hoặc phá hủy cơ sở dữ liệu.
- Cấu hình Bảo mật Sai: Lỗi này xuất hiện khi cài đặt và cấu hình hệ thống không được thực hiện đúng cách, làm cho hệ thống trở nên dễ bị tấn công.
- Vấn đề Chứng thực: Các lỗi xác thực có thể cho phép kẻ tấn công xâm nhập vào hệ thống bằng cách vượt qua cơ chế xác thực hoặc sử dụng thông tin xác thực không hợp lệ.
- Lộ thông tin nhạy cảm: Nếu thông tin nhạy cảm được lưu trữ hoặc truyền đi mà không được mã hóa đúng cách, nó có thể dễ bị lộ ra ngoài trừ.

- Tấn công Lừa đảo: Kẻ tấn công cố gắng lừa người dùng để tiết lộ thông tin cá nhân hoặc thông tin đăng nhập bằng cách mô phỏng các trang web hoặc trang đăng nhập giả mạo. Để ngăn chặn lỗi bảo mật, quan trọng để thực hiện kiểm thử bảo mật định kỳ, giữ phần mềm và hệ thống luôn được cập nhật, áp dụng các biện pháp bảo mật tốt nhất, và đào tạo người sử dụng để phòng tránh các hành vi rủi ro.

5. Lỗi Tương Thích:

Lỗi tương thích xuất hiện khi một phần mềm hoặc hệ thống không tương thích hoặc không chạy đúng trên một môi trường hay nền tảng cụ thể. Đây có thể là một vấn đề quan trọng khi phần mềm được triển khai trên nhiều hệ điều hành, trình duyệt, hoặc môi trường khác nhau. Dưới đây là một số ví dụ về lỗi tương thích:

- Tương Thích Hệ Điều Hành:

Một ứng dụng được phát triển cho hệ điều hành Windows có thể không hoạt động đúng trên hệ điều hành macOS hoặc Linux, và ngược lại.

Các phần mềm sử dụng tính năng hoặc thư viện chỉ có sẵn trên một hệ điều hành cụ thể.

- Tương Thích Trình Duyệt:

Trang web hoặc ứng dụng web không hiển thị đúng trên các trình duyệt khác nhau (ví dụ: Internet Explorer, Chrome, Firefox).

Sự khác biệt trong việc hỗ trợ các tiêu chuẩn web có thể dẫn đến lỗi trên các trình duyệt khác nhau.

- Tương Thích Phiên Bản:

Một ứng dụng yêu cầu một phiên bản cụ thể của một thư viện hay framework, nhưng người dùng cài đặt phiên bản khác.

Phiên bản của cơ sở dữ liệu không tương thích với phiên bản của ứng dụng.

- Tương Thích Hardware:

Một ứng dụng yêu cầu tính năng hay tài nguyên từ phần cứng không có sẵn trên một số thiết bị hoặc máy tính.

Ứng dụng sử dụng các chỉ thị xử lý không tương thích với loại CPU trên một máy tính cụ thể.

- Tương Thích Ngôn Ngữ:

Một ứng dụng được viết bằng một ngôn ngữ lập trình cụ thể có thể không tương thích với môi trường chạy hay thư viện hỗ trợ trên một ngôn ngữ khác.

Để giải quyết lỗi tương thích, nhóm phát triển cần thực hiện kiểm thử kỹ thuật và chắc chắn rằng ứng dụng của họ hoạt động đúng trên nhiều môi trường khác nhau. Các bản cập nhật và sửa lỗi có thể cần được triển khai để cải thiện tương thích với các hệ điều hành, trình duyệt, và phiên bản khác nhau.

6. Lỗi Tích Hợp:

Lỗi tích hợp xuất hiện khi các thành phần khác nhau của một hệ thống không hoạt động chính xác khi được kết hợp với nhau. Các hệ thống phần mềm thường bao gồm nhiều thành phần, modules, hoặc dịch vụ khác nhau, và lỗi tích hợp có thể xảy ra khi chúng không tương tác đúng cách. Dưới đây là một số ví dụ về lỗi tích hợp.

- Thiếu giao tiếp:

- Hệ thống sử dụng giao thức hoạt động cổng giao tiếp không đúng cách, dẫn đến việc các thành phần không thể trao đổi thông tin.
- Lỗi trong việc chuyển dữ liệu khi giữa các phần.

- Chồng Chát Dữ Liệu(Data):

- Các thành phần gửi và nhận dữ liệu với cấu trúc khác nhau, dẫn đến sự hiểu lầm hoặc mất dữ liệu.
- Trùng lặp dữ liệu khi nhiều thành phần cùng truy cập một nguồn dữ liệu chung.

- Lỗi Gọi Hàm:

- Gọi hàm với tham số không đúng hoặc kiểu dữ liệu không phù hợp.
 - Sai cú pháp trong cách các hàm được gọi và trả giá trị.
- Xung Đột Tên:
- Các thành phần có các tên trùng nhau, gây hiểu lầm khi chúng được sử dụng trong hệ thống tích hợp.
 - Xung đột giữa các namespace hoặc tên biến.
- Không Đồng Bộ (Asynchronous Issues):

Lỗi xảy ra khi có các sự kiện xảy ra không đồng bộ giữa các thành phần.

Ví dụ: Các tác vụ được thực hiện ở nền không hoàn thành trước khi cần chúng.

- Không Đồng Bộ (Asynchronous Issues):

Lỗi xảy ra khi có các sự kiện xảy ra không đồng bộ giữa các thành phần.

Ví dụ: Các tác vụ được thực hiện ở nền không hoàn thành trước khi cần chúng.

- Lỗi Thứ Bậc (Hierarchy Errors):
- Các lỗi xảy ra khi có sự hiểu lầm về cấp độ hoặc thứ bậc của các thành phần trong hệ thống.
 - Sự không đồng bộ giữa các thành phần cấp độ cao và cấp độ thấp.

Giải quyết lỗi tích hợp thường đòi hỏi sự chặt chẽ trong quá trình kiểm thử, đặc biệt là kiểm thử tích hợp, để đảm bảo rằng tất cả các thành phần tương tác với nhau đúng cách trong môi trường hệ thống thực tế.

PHẦN V: KẾT LUẬN

- Kiểm thử phần mềm hiện nay là vấn đề rất quan trọng với các tổ chức phát triển phần mềm. Trong bài báo cáo đồ án của em do còn thiếu thời gian và kinh nghiệm còn hạn chế nên phần mềm đoán lỗi em chưa thể nghiên cứu sâu về nó.
- Thời gian thực hiện báo cáo đồ án cơ sở ngành dưới sự hướng dẫn của giáo viên Nguyễn Khắc Quốc bài báo cáo của em đã thực hiện tốt mục tiêu sao.

1. Kết quả đạt được

- Trình bài đầy đủ và tổng quan về các phần mềm, công nghệ phần mềm, lỗi của phần mềm, các vấn đề liên quan đến kiểm đoán thử lỗi
- Giới thiệu công cụ kiểm thử Selenium

2. Hạn chế

- Trong thời gian qua em đã cố gắng làm tốt đề tài mà em đã chọn nhưng với kinh nghiệm còn non nớt nên bài báo cáo đồ án của em còn nhiều thiếu sót
- Chỉ đi vào nghiên cứu Selenium

3. Hướng phát triển đề tài

Trong thời gian tới em sẽ tiếp tục nghiên cứu về các vấn đề của kiểm thử đoán lỗi và đặc biệt là kiểm thử trên web Selenium để có thể vận dụng vào các ứng dụng kiểm thử khác trong quá trình sắp tới để cải thiện mức độ công việc tốt nhất có thể.

TÀI LIỆU THAM KHẢO

1. Tìm hiểu về kiểm thử website <https://s.net.vn/eNgo>
2. Quy trình kiểm thử website <https://s.net.vn/Ghdf>
3. Mức độ kiểm thử website <https://s.net.vn/0obg>
4. Kiểm thử hộp đen <https://s.net.vn/qSol>
5. Kiểm thử hộp trắng <https://s.net.vn/CnLx>
6. Tổng quan về Selenium <https://s.net.vn/RFdv>