



KIỂM THỬ TỰ ĐỘNG HƯỚNG DẪU DỮ LIỆU CHO ỨNG DỤNG WEB SỬ DỤNG SELENIUM WEBDRIVER

Đỗ Thị Thu Trang, Lê Thị Linh, Trịnh Thị Nhị, Ngô Thanh Huyền
Trường Đại học Sư phạm Kỹ thuật Hưng Yên

Ngày tòa soạn nhận được bài báo: 20/06/2018

Ngày phân biện đánh giá và sửa chữa: 06/08/2018

Ngày bài báo được duyệt đăng: 13/08/2018

Tóm tắt:

Kiểm thử tự động hướng dữ liệu sử dụng dữ liệu tách rời với kịch bản kiểm thử, lưu trữ trong các dạng bảng tính sử dụng rất hiệu quả trong việc kiểm thử ứng dụng Web. Mục tiêu của bài báo này là xây dựng một công cụ kiểm thử tự động hướng dữ liệu cho ứng dụng Web sử dụng nền tảng Selenium WebDriver. Với công cụ kiểm thử này, người sử dụng có thể kiểm thử bất kỳ trang Web nào mà không cần trang bị kiến thức về lập trình.

Từ khóa: Kiểm thử phần mềm, Kiểm thử tự động, WebDriver, Framework, Kiểm thử hướng dữ liệu.

1. Giới thiệu

Ngành Công nghiệp Phần mềm đang tăng trưởng theo cấp số nhân và tạo ra rất nhiều ứng dụng phần mềm hữu ích, trong đó có ứng dụng Web - loại ứng dụng đang được sử dụng rất rộng rãi và phổ biến hiện nay [3]. Ứng dụng Web ngày nay không chỉ ra tăng gia về số lượng mà còn gia tăng về yêu cầu chất lượng và độ phức tạp của ứng dụng.

Thực tế cho thấy, việc thực hiện kiểm thử thủ công cho các ứng dụng phần mềm nói chung và ứng dụng Web nói riêng bộc lộ nhiều nhược điểm như: thời gian thực hiện kiểm thử kéo dài, công việc lặp đi lặp lại gây nhàm chán, một số lỗi không thể kiểm thử thủ công, gặp khó khăn khi thực hiện kiểm thử hồi quy [11].

Điều này đặt ra vấn đề cần phải có một cách thức kiểm thử để khắc phục các nhược điểm trên của kiểm thử thủ công. Vì vậy, kiểm thử tự động ra đời. Việc sử dụng kiểm thử tự động trong kiểm thử phần mềm là một bước tiến quan trọng và cần thiết trong quy trình kiểm thử phần mềm.

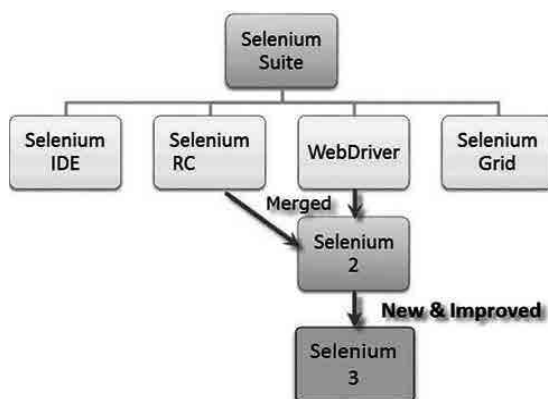
Dựa trên các yêu cầu kiểm thử của phần mềm, kiểm thử tự động chia thành các hướng phát triển khác nhau như: kiểm thử hướng mô hình (Modular Testing), kiểm thử hướng dữ liệu (Data Driven Testing), kiểm thử hướng từ khóa (Keyword Driven Testing)... [12]. Trong đó, kiểm thử hướng dữ liệu là hướng kiểm thử được sử dụng phổ biến nhất trong quá trình kiểm thử các ứng dụng Web.

Kiểm thử hướng dữ liệu [6] là cách kiểm thử có thể giảm bớt thời gian kiểm thử đối với các trường hợp sử dụng cùng các bước thực hiện kiểm thử nhưng mỗi kịch bản kiểm thử chỉ khác nhau ở bộ dữ liệu đầu vào. Với hướng tiếp cận này, dữ liệu kiểm thử (Test Data) được tách khỏi kịch bản kiểm thử (Test Script), lưu trữ dưới dạng các file

dữ liệu như XML, Excel, CSV, Database... Các dữ liệu này được lưu trữ theo một quy ước chung là 'Key - Value', các từ khóa (Key) này sẽ được sử dụng để truy cập và truyền dữ liệu (Value) vào các kịch bản kiểm thử tương ứng thông qua một số thư viện chung [2].

Mặc dù có nhiều framework hỗ trợ viết kịch bản kiểm thử tự động cho ứng dụng Web, tuy nhiên Selenium được đánh giá là framework kiểm thử tự động có nhiều ưu thế hơn cả như: không mất chi phí bản quyền, thực hiện kiểm thử tự động trên nhiều hệ điều hành, hỗ trợ kiểm thử nhiều trình duyệt, hỗ trợ nhiều ngôn ngữ lập trình [13].

Selenium [5] là một bộ công cụ kiểm thử mã nguồn mở bao gồm 4 thành phần: Selenium Integrated Development Environment (IDE), Selenium Remote Control (RC), WebDriver và Selenium Grid.



Hình 1. Bộ công cụ Selenium [5]

Mỗi công cụ đáp ứng một nhu cầu kiểm thử khác nhau của ứng dụng Web. Nó thực sự mạnh mẽ khi so sánh với các công cụ kiểm thử tự động khác.

Selenium rất linh hoạt và dễ sử dụng. Thêm nữa, ngôn ngữ lập trình xây dựng chương trình kiểm thử được độc lập với ngôn ngữ xây dựng ứng dụng Web. Selenium là framework mã nguồn mở được đánh giá là tốt nhất hiện nay [16].

Trong đó, WebDriver là một framework được thiết kế để hỗ trợ lập trình nâng cao với các kịch bản kiểm thử ứng dụng Web. Chúng ta có thể xây dựng công cụ kiểm thử đặc thù dựa trên nền tảng hỗ trợ của Selenium WebDriver.

Theo khảo sát của chúng tôi thì hiện nay trên thị trường có khá nhiều công cụ kiểm thử tự động hỗ trợ kiểm thử hướng dữ liệu như UFT (Unified Functional Testing) [9], Katalon Studio [10], TestComplete [14]. Tuy nhiên, những công cụ này thì đòi hỏi người sử dụng phải có kiến thức nhất định về lập trình hoặc phải chi trả một giá phí rất cao để được sử dụng.

Bài báo này nghiên cứu và xây dựng một công cụ kiểm thử hướng dữ liệu cho ứng dụng Web dựa trên nền tảng Selenium WebDriver. Chúng tôi lựa chọn xây dựng công cụ này vì những lý do như sau:

- Ứng dụng Web hiện nay đang trở lên rất phổ biến, việc kiểm thử tự động ứng dụng Web là vô cùng cần thiết.

- Ứng dụng Web ngày càng trở nên phức tạp, việc kiểm thử thủ công là điều không thể. Chúng tôi xây dựng một công cụ có thể kiểm thử chức năng

hướng dữ liệu cho bất kỳ trang web nào và kiểm thử trên bất kỳ trình duyệt phổ biến nào.

- Công cụ chúng tôi xây dựng không yêu cầu kiểm thử viên phải có kiến thức về lập trình.

Phần còn lại của bài báo có cấu trúc như sau: phần 2 sẽ giới thiệu về mô hình kiểm thử hướng dữ liệu và framework kiểm thử tự động Selenium WebDriver, phần 3 sẽ mô tả bài toán, chương trình cài đặt và kết quả đạt được. Phần cuối cùng là kết luận.

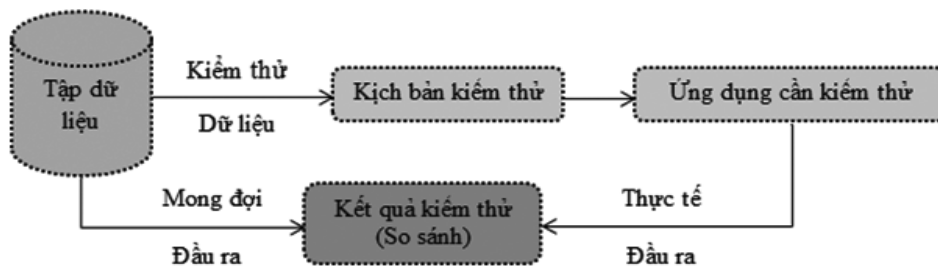
2. Mô hình

2.1. Mô hình kiểm thử hướng dữ liệu

Kiểm thử tự động hướng dữ liệu là tạo ra các kịch bản kiểm thử nơi dữ liệu kiểm thử hoặc các giá trị đầu ra được đọc từ các tệp dữ liệu thay vì sử dụng các giá trị cụ thể cho một lần chạy kiểm thử. Bằng cách này, kiểm thử viên có thể kiểm tra các ứng dụng xử lý các đầu vào khác nhau một cách hiệu quả. Dữ liệu có thể lưu ở một trong các kiểu file dữ liệu như: XLS, XML, CSV, DataBase.

Với cách xây dựng kịch bản kiểm thử này thì kịch bản kiểm thử sẽ cung cấp logic kiểm thử có thể sử dụng lại để giảm thời gian, công sức khi bảo trì và tăng phạm vi kiểm tra. Các giá trị dữ liệu đầu vào và kết quả mong đợi (theo các tiêu chí kiểm tra) có thể được lưu trữ trong một hoặc nhiều nguồn dữ liệu hoặc cơ sở dữ liệu, định dạng và tổ chức thực tế tùy thuộc yêu cầu thực hiện cụ thể.

2.2. Nguyên lý hoạt động



Hình 2. Mô hình kiểm thử tự động hướng dữ liệu [7]

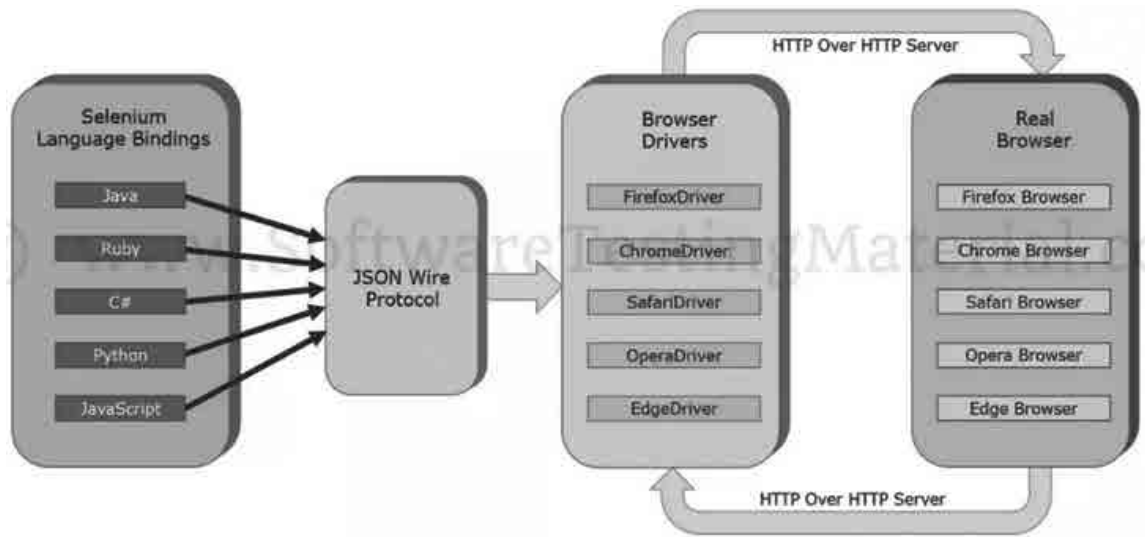
Thông thường, tập dữ liệu bao gồm các giá trị đầu vào và kết quả mong đợi đầu ra.

Khi bộ nhớ chứa cả dữ liệu đầu vào và kết quả mong đợi đầu ra, kiểm thử theo hướng dữ liệu bao gồm các hoạt động sau được thực hiện trong vòng lặp:

- Lấy dữ liệu đầu vào từ nguồn dữ liệu
- Nhập dữ liệu tự động vào ứng dụng cần kiểm thử thông qua kịch bản kiểm thử.
- Thực hiện so sánh kết quả thực tế sau khi thực thi kiểm thử ứng dụng với kết quả mong đợi trong tập dữ liệu.
- Tiếp tục vòng lặp với bộ dữ liệu đầu vào tiếp theo.

2.3. Selenium WebDriver

Selenium WebDriver [7] là một framework mã nguồn mở giúp xây dựng chương trình kiểm thử thực thi các hành động lên trang Web một cách tự động. WebDriver chạy trực tiếp trên trình duyệt và hỗ trợ hầu như tất cả các trình duyệt phổ biến như Firefox, Chrome, IE, Opera, Safari. WebDriver chạy trên tất cả các nền tảng như Windows, Linux và Macintosh. Đây là một công cụ rất hữu ích cho việc kiểm thử chức năng của ứng dụng Web và kiểm tra khả năng tương thích với trình duyệt. WebDriver hỗ trợ viết kịch bản kiểm thử trên nhiều ngôn ngữ khác nhau bao gồm Java, C#, Python, PHP, Ruby, Perl. Về mặt bản chất, WebDriver là một gói cung cấp các API hay gọi là thư viện.



Hình 3. Kiến trúc Selenium WebDriver [15]

3. Bài toán và chương trình

3.1. Bài toán

Bài toán đặt ra là xây dựng một công cụ kiểm thử chức năng hướng dữ liệu thực hiện kiểm thử được trên nhiều trang Web khác nhau và trên nhiều trình duyệt khác nhau như Google Chrome, FireFox, IE, Opera, Safari mà không yêu cầu người sử dụng cần trang bị kiến thức về lập trình.

Vấn đề đặt ra là mỗi trang Web có số lượng đối tượng cần tương tác trên trang là khác nhau, dẫn đến số lượng đầu vào kiểm thử là khác nhau. Như vậy cần xây dựng phần mềm với số lượng đầu vào là tùy biến. Đồng thời một đối tượng trên trang Web có thể xác định được bằng nhiều cách khác nhau dựa trên mã HTML của trang Web đó, bao gồm các kiểu định vị như: Id, Name, Class Name, XPath,

CssSelector, LinkText, Partial Link Text, Tag name.

Input: tập dữ liệu đầu vào bao gồm các thông tin: mã trường hợp kiểm thử, kiểu định vị của từng đối tượng trên trang, giá trị định vị của từng đối tượng, giá trị dữ liệu cần nhập cho đối tượng đó và kết quả mong đợi tương ứng với từng trường hợp kiểm thử.

Output: kết quả của một trường hợp kiểm thử nhận giá trị Pass hoặc Fail. Hỗ trợ báo cáo thông qua email, xuất tệp Excel và chụp ảnh màn hình lúc thực thi kịch bản.

3.2. Dữ liệu

Dữ liệu trong bài này chúng tôi xây dựng thử nghiệm cho chức năng tạo tài khoản của Google. Tập dữ liệu cho kịch bản kiểm thử có cấu trúc như sau:

Bảng 1. Cấu trúc testcase hướng dữ liệu

TC ID	Locator 1	Locator 2	Locator 3	Locator 4	Locator 5	Button	Value 1	Value 2	Value 3	Value 4	Value 5	Act_Loc	Exp
TC0 1	Id@firstName	Name@lastName	Id@username	CssSelector@#password>div.aCsJod.oJeWuf > div > div.Xb9hP > input	Name@confirmPassword	XPath@//*[@id="accountDetailsNext"]/content/span		Linh	Linh96utehy	123456	123456	XPath@//*[@id="view_container"]/form/div[2]div/div[1]div/div[1]div/div[2]/div[2]	Enter first name
TC0 2	Id@firstName	Name@lastName	Id@username	CssSelector@#password>div.aCsJod.oJeWuf > div > div.Xb9hP > input	Name@confirmPassword	XPath@//*[@id="accountDetailsNext"]/content/span	Lê	Linh411utehy	Lethilinh	Lethilinh	Lethilinh	XPath@//*[@id="view_container"]/form/div[2]div/div[1]div/div[2]div/div[2]/div[2]	Enter last name
TC0 3	Id@firstName	Name@lastName	Id@username	CssSelector@#password>div.aCsJod.oJeWuf > div > div.Xb9hP > input	Name@confirmPassword	XPath@//*[@id="accountDetailsNext"]/content/span	Lê	Linh	Linhle	Linh2411	Linh2411	XPath@//*[@id="headingText"]	Verify your phone number

Cấu trúc tệp dữ liệu bao gồm các phần:

- TCID: là mã của một trường hợp kiểm thử. Mỗi một dòng tương ứng với một trường hợp kiểm thử.

- Locator 1- Locator 5: dùng để xác định các đối tượng trên trang Web cần tương tác khi kiểm thử bao gồm 2 phần: phần trước dấu @ là các từ khóa bao gồm: Id, Name, Class Name, XPath, CssSelector, LinkText, Partial Link Text, Tag name; sau dấu @ là giá trị của cách định vị tương ứng. Tùy thuộc số đối tượng trên trang Web cần tương tác, số lượng các trường Locator có thể tăng hoặc giảm. Tùy thuộc vào mã HTML mà trang Web được xây dựng, các từ khóa xác định đối tượng được lựa chọn là khác nhau.

- Button: xác định đối tượng là Button trên trang Web. Cách viết tương tự như Locator phía trên.

- Value 1 - Value 5: giá trị đầu vào tương ứng với các đối tượng được xác định bởi các Locator trên. Mỗi một dòng là một bộ giá trị đầu vào tương ứng cho một trường hợp kiểm thử. Mỗi bộ dữ liệu đầu vào được xác định thông qua các kỹ thuật kiểm thử hộp đen như: Phân vùng tương đương, Phân tích giá trị biên, Kỹ thuật bảng quyết định [8].

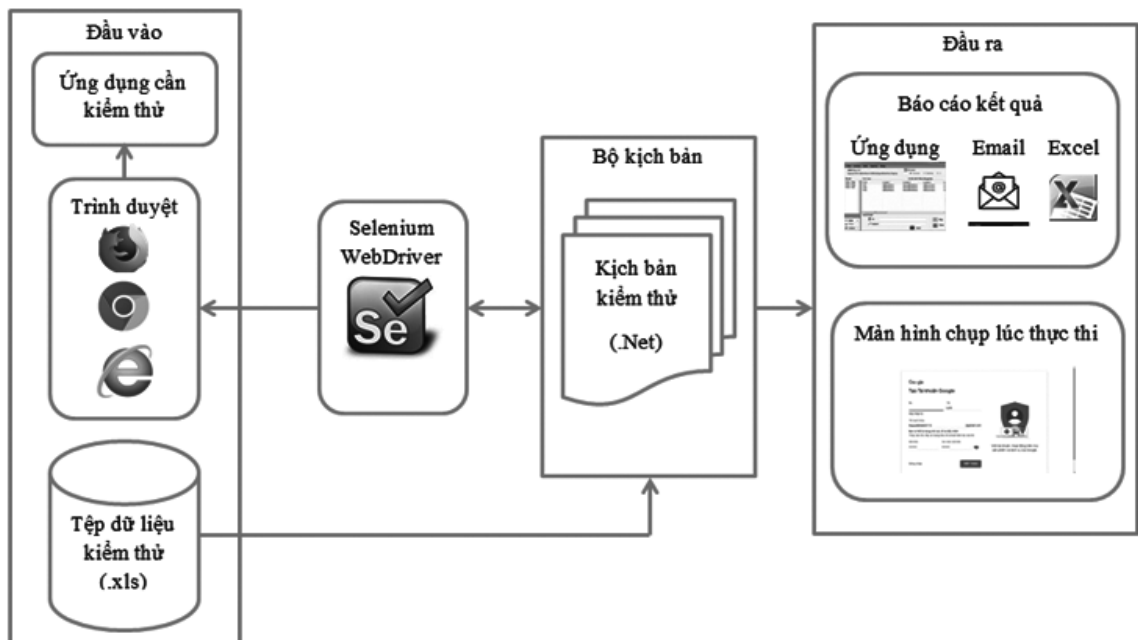
- Act_Loc: xác định vị trí của đối tượng trên trang chứa kết quả thực tế sau khi thực hiện kịch bản kiểm thử.

- Exp: kết quả mong đợi của trường hợp kiểm thử.

3.3. Xây dựng ứng dụng kiểm thử

Trong phần này chúng tôi sẽ mô tả chi tiết nguyên lý hoạt động và các chương trình cài đặt chính của công cụ.

Nguyên lý hoạt động



Hình 4. Nguyên lý hoạt động của công cụ

Hoạt động của công cụ kiểm thử tự động hướng dữ liệu này có thể phân tích thành 3 bước chính như sau:

- **Bước 1:** Nhập các yêu cầu đầu vào cho chương trình bao gồm: (1) Ứng dụng cần kiểm thử: Đường link của trang Web cần kiểm thử (Application Under Test). (2) Trình duyệt: lựa chọn trình duyệt thực hiện kiểm thử. (3) Tệp dữ liệu kiểm thử: Nhập các trường hợp kiểm thử và dữ liệu kiểm thử có cấu trúc được xây dựng như Bảng 1.

- **Bước 2:** Xử lý từng trường hợp kiểm thử và kiểm thử tự động ứng dụng Web thông qua kịch

bản kiểm thử được xây dựng và dựa trên nền tảng hỗ trợ của Selenium WebDriver. Chương trình thực hiện tự động các thao tác kiểm thử trên trình duyệt được chọn với từng dòng dữ liệu trong tệp dữ liệu: nhập dữ liệu vào các đối tượng trên trang thông qua các Locator và Value; nhấn nút Button trên trang được xác định bởi Button và so sánh kết quả thực tế với kết quả mong đợi Exp. Kết quả thực tế được xác định thông qua cột dữ liệu Act_Loc.

- **Bước 3:** Đưa ra kết quả kiểm thử qua ba kênh: trên giao diện công cụ, gửi email và xuất ra excel. Chương trình hỗ trợ có thể gửi mail tới

nhiều người liên quan cùng một lúc. Kết quả kiểm thử được xuất ra tệp Excel giúp cho kiểm thử viên có thể tổng hợp báo cáo một cách dễ dàng, nhanh chóng và thuận tiện hơn.

Ngoài ra chương trình còn cung cấp tính năng: Chụp ảnh màn hình lúc thực thi kiểm thử giúp kiểm thử viên dễ dàng trong việc tìm lỗi hơn.

Các chương trình chính được xây dựng:

Chương trình 1: Xử lý tệp dữ liệu đầu vào:

Bước 1: Thực hiện chuyển dữ liệu từ tệp Excel sang dạng list. Chúng ta sẽ thao tác với dữ liệu trên list.

Bước 2: Sử dụng vòng lặp thực hiện các công việc sau:

Xét dòng đầu tiên của list, chính là tiêu đề của cột dữ liệu:

Nếu bắt đầu bởi từ “Locator”, “Button”, “Atc” thì xét các ký tự đầu trước phần @ trong ô dữ liệu. Các ký tự đầu là ký hiệu của 8 kiểu định vị đối tượng trên trang Web được xác định bởi Selenium WebDriver: Id, Name, Class Name, XPath, CssSelector, LinkText, Partial Link Text, Tag name. Dựa vào kiểu định vị, ứng dụng sẽ tương tác với đối tượng trên trang Web.

Nếu bắt đầu bởi từ “Value” thì lấy giá trị nhập vào cho đối tượng trên trang Web tương ứng.

Nếu là “Act_Loc” dùng để lấy ra kết quả thực tế sau khi thực thi kịch bản kiểm thử với một bộ dữ liệu cụ thể.

Nếu là “Exp” thì đưa ra kết quả mong đợi tương ứng cho kịch bản kiểm thử đó.

Bước 3: Thực hiện so sánh kết quả thực tế với kết quả mong đợi:

Nếu kết quả thực tế và kết quả mong đợi giống nhau thì trả về kết quả là Pass. Đồng thời cộng kết quả số lượng trường hợp kiểm thử Pass thêm 1 đơn vị.

Nếu kết quả thực tế và kết quả mong đợi không giống nhau thì trả về kết quả là Fail. Đồng thời cộng kết quả số lượng trường hợp kiểm thử Fail thêm 1 đơn vị.

Tổng số trường hợp kiểm thử được tính dựa trên độ dài của list.

Chương trình 2: Chuyển đổi dữ liệu từ tệp dữ liệu sang dạng list (danh sách).

Bước 1: Tạo đối tượng Excel sử dụng thư viện Microsoft.Office.Interop.Excel.

Bước 2: Sử dụng vòng lặp đọc lần lượt từng dòng dữ liệu chuyển sang list bao gồm cả dòng tiêu đề.

Chương trình 3: Thực hiện gửi email tới các thành viên liên quan.

Bước 1: Khai báo sử dụng dịch vụ Google

SMTP của gmail.

Bước 2: Thực hiện lấy các thông tin từ công cụ bao gồm: Địa chỉ email người gửi, Địa chỉ email người nhận (số lượng email người nhận có thể là một hoặc nhiều người, tùy thuộc vào số email được nhập), Tiêu đề thư, Nội dung thư. Nội dung thư bao gồm các thông tin sau: Tổng số trường hợp kiểm thử, số lượng trường hợp kiểm thử “Pass”, số lượng trường hợp kiểm thử “Fail” và kết quả của từng trường hợp kiểm thử cụ thể. Và tệp đính kèm ảnh giao diện ứng dụng Web lúc thực thi kiểm thử.

Bước 3: Thực hiện cấu hình gửi thư và đưa ra thông báo khi thực hiện gửi thành công.

3.4. Kết quả thực hiện

Sau khi thực thi kịch bản kiểm thử, kết quả thu được như sau:

Bảng 2. Kết quả kiểm thử

Tiêu đề	Kết quả	Ghi chú
Total:	125	Tổng số trường hợp kiểm thử
Pass:	103	Tổng số trường hợp kiểm thử thành công
Fail:	22	Tổng số trường hợp kiểm thử không thành công
TC01	Pass	Kết quả kiểm thử của mã TC01
TC02	Pass	Kết quả kiểm thử của mã TC02
TC03	Fail	Kết quả kiểm thử của mã TC03
...

Từ thông tin ở Bảng 2, chúng ta có một số phân tích: chúng ta có thể tính được tỉ lệ test case đã thành công trên tổng số lượng test case. Từ đó có thể đưa ra quyết định đã dừng kiểm thử được hay chưa. Đồng thời thông tin kết quả cụ thể của từng test case cho phép chúng ta biết trường hợp kiểm thử nào chưa thành công, thông tin này được chuyển cho bên lập trình viên để thực hiện chỉnh sửa. Ngoài ra, việc lưu ảnh chụp màn hình lúc thực thi giúp việc tìm lỗi được nhanh chóng hơn.

Một lợi thế của công cụ so với các công cụ khác là tự động chụp ảnh màn hình ứng dụng Web lúc thực thi kiểm thử và tự động gửi kết quả kiểm thử tới những người liên quan thông qua email.

4. Kết luận

Bài báo này có đóng góp như sau: chúng tôi xây dựng một công cụ kiểm thử tự động hướng dữ liệu có thể kiểm thử chức năng bất kỳ trang Web nào trên ba trình duyệt phổ biến là Chrome, Firefox và IE, đồng thời không yêu cầu người sử dụng có kiến thức lập trình.

Công việc tiếp theo của chúng tôi là mở

rộng kết quả đạt được của bài báo bằng 2 công việc. Thứ nhất, chúng tôi muốn mở rộng đầu vào dữ liệu không chỉ trên file dữ liệu Excel, mà trên nhiều dạng file dữ liệu khác, cụ thể như XML, CSV, Database; mở rộng kiểm thử trên hai trình duyệt là Opera và Safari. Thứ hai, chúng tôi muốn triển khai kiểm thử

trên ứng dụng mobile, cụ thể là ứng dụng Android và IOS.

Sau khi đã tìm hiểu và so sánh như vậy, chúng tôi tin rằng công cụ này có thể được mở rộng và có thể ứng dụng hỗ trợ kiểm thử tự động chức năng hướng dữ liệu một cách hiệu quả.

Tài liệu tham khảo

- [1]. Bruns, A., Kornstadt, A., & Wichmann, D., Web application tests with selenium. *IEEE software*, **26(5)**, 2009.
- [2]. Vina M Lomte, Rishikesh Chandra, Ayush Gondhali, Ashish Shinde, Sanket Pimple, Data Driven Automation Testing Framework. *IJETCAS*, 2014, pp. 51-56.
- [3]. Murugesan, S., Deshpande, Y., Hansen, S., & Ginige, A., Web engineering: A new discipline for development of web-based systems. In *Web Engineering*, Springer, Berlin, Heidelberg, 2001, pp. 3-13.
- [4]. Conallen, J., Modeling Web application architectures with UML. *Communications of the ACM*, 1999, **42(10)**, 63-70.
- [5]. Krishna Rungta, Learn Selenium in 1 Day: Definitive Guide to Learn Selenium for Beginners. Penguin Random House South Afr, 2017.
- [6]. Carl Cocchiaro, Selenium Framework Design in Data-Driven Testing. Packt, 2018.
- [7]. Zhan, Zhimin, Selenium WebDriver Recipes in C#, Apress, 2015.
- [8]. Phạm Ngọc Hùng, Trương Anh Hoàng, Đặng Văn Hưng, *Giáo trình kiểm thử phần mềm*, 2014, pp. 108-130.
- [9]. <https://www.softwaretestinghelp.com/qtp-tutorial-18-data-driven-and-hybrid-frameworks/>
- [10]. https://medium.com/@katalon_studio/data-driven-testing-approach-with-katalon-studio-b835c9e491dd
- [11]. <https://viblo.asia/p/kiem-thu-thu-cong-manual-testing-va-kiem-thu-tu-dong-automated-testing-QWkwGnpER75g>
- [12]. <https://www.softwaretestingmaterial.com/types-test-automation-frameworks/>
- [13]. <https://medium.com/@briananderson2209/best-automation-testing-tools-for-2018-top-10-reviews-8a4a19f664d2>
- [14]. <https://smartbear.com/product/testcomplete/overview/>
- [15]. <https://www.softwaretestingmaterial.com/selenium-webdriver-architecture/>
- [16]. <https://dzone.com/articles/10-best-open-source-test-automation-frameworks-for>

DATA DRIVEN AUTOMATION TESTING FRAMEWORK FOR WEB APPLICATIONS USING SELENIUM WEBDRIVER

Abstract:

Software testing is the main technique to ensure quality and finding bugs. In general, this is a difficult and time-consuming task. Web applications are becoming more and more complex. Due to the peculiarities of such applications, so manual testing is often poorly performed or skipped by testers. Test automation can help to avoid this situation.

Data-driven automated testing with test scripts stored in spreadsheet formats is very effective in testing Web applications. The goal of this paper is to develop a self-guided data-driven testing tool for web applications using the Selenium WebDriver framework. With this testing tool, users can test any web page without programming knowledge.

Keywords: *Software Testing, Automation Testing, Selenium WebDriver, Framework, Data Driven Framework.*