**BỘ CÔNG THƯƠNG**



**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH KHOA CÔNG NGHỆ THÔNG TIN**

**MÔN: ĐẢM BẢO CHẤT LƯỢNG VÀ KIỂM THỬ PHẦN MỀM**

*Đề tài:* TÌM HIỂU VỀ TEST TOOL SELENIUM

Lớp học phần: DHKTPM16CTT

Họ tên sinh viên: Nguyễn Xuân An

Nhóm:

GVHD: Châu Thị Bảo Hà

Thành phố Hồ Chí Minh, ngày 10 tháng 08 năm 2023

Mục lục

[**1.** **Giới thiệu về Selenium** 3](#_Toc142750614)

[**1.1.** **Khái niệm** 3](#_Toc142750615)

[**1.2.** **Ngôn ngữ lập trình hỗ trợ** 3](#_Toc142750616)

[**1.3.** **Các thành phần của Selenium** 3](#_Toc142750617)

[**1.4.** **Các nền tảng hỗ trợ** 4](#_Toc142750618)

[**1.5.** **Link download** 4](#_Toc142750619)

[**1.6.** **Bản quyền** 4](#_Toc142750620)

[**1.7.** 4](#_Toc142750621)

[**2.** **Chức năng** 4](#_Toc142750622)

[**2.1.** **Selenium Remote Control (RC)** 4](#_Toc142750623)

[**2.2.** **Selenium Grid** 5](#_Toc142750624)

[**2.3.** **Selenium IDE (Selenium Integrated Development Environment)** 6](#_Toc142750625)

[**2.4.** 6](#_Toc142750626)

[**3.** **Mục đích của việc kiểm thử tự động** 6](#_Toc142750627)

[**3.1.** **Tiết kiệm thời gian và tài nguyên** 6](#_Toc142750628)

[**3.2.** **Tự động hóa việc kiểm thử lặp đi lặp lại** 6](#_Toc142750629)

[**3.3.** **Tự động hóa các trường hợp kiểm thử phức tạp** 6](#_Toc142750630)

[**3.4.** 6](#_Toc142750631)

[**4.** **Ưu và nhược điểm của Selenium** 7](#_Toc142750632)

[**5.** **Cấu trúc của Selenium** 7](#_Toc142750633)

[**6.** **Tìm hiểu bộ công cụ của Selenium** 7](#_Toc142750634)

[**7.** **Cài đặt Selenium** 7](#_Toc142750635)

[**8.** **Demo** 7](#_Toc142750636)

[**Danh mục tài liệu tham khảo** 7](#_Toc142750637)

[**Chú thích:** 7](#_Toc142750638)

1. **Giới thiệu về Selenium**
   1. **Khái niệm**

Selenium là một bộ công cụ kiểm thử mã nguồn mở được sử dụng rộng rãi để tự động hóa kiểm thử cho các ứng dụng web. Nó cho phép các nhà phát triển và kiểm thử viên tạo thực thi các kịch bản kiểm thử tự động để đảm bảo rằng các ứng dụng web hoạt động đúng như mong đợi trên nhiều nền tảng và trình duyệt khác nhau.

Vì Selenium là một bộ các công cụ khác nhau nên nó cũng có nhiều nhà phát triển khác nhau. Dưới đây là những người quan trọng đã đóng góp vào dự án Selenium:

Ban đầu Selenium được tạo ra bởi Jason Huggins vào năm 2004. Ông là một kỹ sư tại ThoughtWorks, và đang làm việc trên một ứng dụng web yêu cầu thử nghiệm thường xuyên. Nhận thấy rằng việc kiểm trả thủ công lặp đi lặp lại của họ ngày càng trở nên không hiệu quả, ông đã tạo ra một chương trình JavaScript sẽ tự động kiểm soát các hành động của trình duyệt. Ông đặt tên cho chương trình này là “*JavaScriptTestRunner.*”

Nhận thấy được tìm năng cho ý tưởng của mình, ông đã làm cho *JavaScriptRunner* trở thành mã nguồn mở, sau này được đổi tên thành Selenium Core.

Vì sao nó có tên gọi là Selenium?

Cái tên “Selenium” xuất phát từ một trò đùa của Jason khi một lần nói với nhóm của mình. Trong quá trình phát triển Selenium, có một framework kiểm thử khác phổ biến được ra bởi công ty có tên Mercury Interactive (công ty đã tạo ra QTP trước khi được HP mua lại). Vì Selenium là một loại thuốc giải nổi tiếng cho ngộ độc Thủy ngân, Json đã gợi ý cái teen đó và được đồng đội chấp nhận. Đó là cách chúng ta gọi framework này cho đến nay.

* 1. **Ngôn ngữ lập trình hỗ trợ**

Selenium hỗ trợ nhiều ngôn ngữ lập trình như Java, Python, C#, Ruby, JavaScript (sử dụng WebDriverIO), …

* 1. **Các thành phần của Selenium**

Selenium bao gồm một bộ các công cụ hỗ trợ kiểm tra tự động tính năng của ứng dụng web:

* Selenium IDE (Selenium Integrated Development Environment)
* Selenium Remote Control (RC)
* Selenium WebDriver
* Selenium Gird
  1. **Các nền tảng hỗ trợ**

Selenium cho phép thực hiện kiểm thử trên nhiều hệ điều hành như Window, macOS, Linux. Ngoài ra, Selenium còn chạy được trên nhiều trình duyệt khác nhau như Chrome, Firefox, Edge, Safari, …

* 1. **Link download**
* <https://www.selenium.dev/downloads/>
  + Link tải các gói ngôn ngữ client driver tương ứng với ngôn ngữ bạn muốn sử dụng để viết các kịch bản tự động hoá.
* Selenium IDE: <https://www.selenium.dev/selenium-ide/>
  1. **Bản quyền**
* Selenium là một dự án mã nguồn mở, cho phép các nhà phát triển tạo và thực hiện các kịch bản tự động hóa cho ứng dụng web. Mã nguồn mở có nghĩa là mã nguồn của Selenium có sẵn cho cộng đồng để sử dụng, tùy chỉnh và đóng góp.

1. **Chức năng**
   1. **Selenium Remote Control (RC)**

* Đối với Selenium Core, các nhà kiểm thử phải cài đặt toàn bộ ứng dụng đang được thử nghiệm và máy chủ web trên máy tính cục bộ của chính họ do các hạn chế được áp đặt bởi chính sách cùng nguồn gốc (same origin policy[1]). Vì vậy, một kỹ sư khác của ThoughtWorks, là Paul Hammant, đã quyết định tạo ra một máy chủ sẽ hoạt động như một proxy HTTP để “đánh lừa” trình duyệt tin rằng Selenium Core và ứng dụng web đang được kiểm thử đến từ cùng một miền. Hệ thống này được biết đến với tên gọi là Selenium Remote Control hoặc Selenium 1.
* Hiện tại, Selenium RC và WebDriver đã được hợp nhất thành một khung công cụ duy nhất để tạo nên Selenium 2. Còn Selenium RC thường được gọi là Selenium 1. Hiện tại, Selenium RC vẫn đang được phát triển nhưng chỉ ở chế độ bảo trì.
* RC có thể hỗ trợ các ngôn ngữ lập trình sau: Java, C#, PHP, Python, Ruby, Perl.
* Là một framework kiểm thử tự động cho ứng dụng web. Selenium hỗ trợ trong công việc kiểm thử như sau:
  + Tự động hóa thao tác người dùng: Selenium RC cho phép bạn viết mã kiểm thử tự động để tương tác với trình duyệt web giống như cách người dùng thực hiện. Điều này bao gồm việc điều khiển trình duyệt để click vào các phần tử, điền thông tin vào các trường nhập liệu, chọn các tùy chọn trong các menu thả xuống, và thậm chí là kiểm tra nội dung hiển thị.
  + Kiểm thử tích hợp: Selenium RC cho phép kiểm thử tích hợp (integration testing) bằng cách tự động thực hiện các kịch bản kiểm thử trên các trang web thực sự. Điều này giúp đảm bảo rằng các thành phần của ứng dụng hoạt động cùng nhau như mong đợi khi được tích hợp lại với nhau.
  + Kiểm thử chức năng: Bạn có thể sử dụng Selenium RC để kiểm tra chức năng của ứng dụng web, bao gồm việc xác định xem các chức năng có hoạt động đúng như dự kiến hay không. Ví dụ, bạn có thể viết mã kiểm thử để đảm bảo rằng việc đăng nhập, đăng ký tài khoản, thêm sản phẩm vào giỏ hàng, và thanh toán đều hoạt động một cách chính xác.
  + Kiểm thử đa trình duyệt: Selenium RC cho phép bạn kiểm tra trên nhiều trình duyệt khác nhau một cách tự động, bao gồm các trình duyệt phổ biến như Chrome, Firefox, Internet Explorer, và Safari. Điều này giúp đảm bảo rằng ứng dụng hoạt động một cách nhất quán trên các nền tảng khác nhau.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| Nhiều trình duyệt và đa nền tảng. | Cài đặt phức tạp. |
| Có thể hỗ trợ việc lặp lại hoặc câu lệnh có điều kiện cho tập lệnh kiểm tra. | Yêu cầu phải có kiến thức về lập trình. |
| Có thể hỗ trợ kiểm tra theo dữ liệu. | Cần phải có Selenium RC server để chạy. |
| Đã hoàn chỉnh API. | API có lệnh thừa và khó hiểu. |
| Có thể sẵn sàng hỗ trợ các trình duyệt mới. | Tương tác với trình duyệt ít hơn thực tế. |
| Thực hiện nhanh hơn Selenium IDE. | Kết quả không phù hợp và sử dụng JavaScript. |
|  | Thời gian thực thi chậm hơn WebDriver. |

* 1. **Selenium Grid**
* Selenium Grid được Patrick Lightbody phát triển để giải quyết nhu cầu tối thiểu hóa thời gian thực hiện kiểm thử càng nhanh càng tốt. Ban đầu ông đặt tên cho hệ thống này là "Hosted QA" (Kiểm thử chất lượng đặt trên máy chủ). Hệ thống này có khả năng chụp ảnh màn hình trình duyệt trong các giai đoạn quan trọng và cũng có khả năng gửi lệnh Selenium đến các máy khác nhau cùng một lúc.
* Selenium Grid là một công cụ được sử dụng cùng với Selenium RC để chạy các bài kiểm tra song song trên các máy tính và trình duyệt khác nhau cùng một lúc. Có nghĩa là chạy nhiều bài kiểm tra cùng một lúc.
* Selenium Grid có các đặc điểm sau đây:
  + Cho phép chạy kiểm tra đồng thời trên nhiều trình duyệt và môi trường khác nhau.
  + Tiết kiệm thời gian một cách đáng kể.
  + Sử dụng khái niệm hub và nodes. Trung tâm điều khiển (hub) hoạt động như một nguồn lệnh Selenium trung tâm gửi đến mỗi nút (node) được kết nối với nó.
  1. **Selenium IDE (Selenium Integrated Development Environment)**
* Shinya Kasatani của Nhật Bản đã tạo ra Selenium IDE, một tiện ích mở rộng cho trình duyệt Firefox và Chrome có khả năng tự động hóa trình duyệt thông qua tính năng ghi và phát lại. Ông đã nảy ra ý tưởng này để tăng tốc độ tạo các trường hợp kiểm thử. Ông đã quyên góp Selenium IDE cho Dự án Selenium vào năm 2006.
  + Là framework đơn giản và dễ học.
  + Là một plugin của Firefox mà bạn có thể dễ dàng cài đặt như các plugin khác.
  + Vì đơn giản mà Selenium IDE chỉ nên được sử dụng như là một công cụ tạo mẫu. Sử dụng Selenium RC và WebDriver để có thể tạo các trường hợp kiểm thử nâng cao hơn.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| Dễ cài đặt và sử dụng. | Thiết kế chỉ để tạo mẫu của kiểm thử. |
| Không yêu cầu kinh nghiệm lập trình, chỉ cần có kiến thức HTML và DOM. | Không cung cấp sự lập lại hoặc câu lệnh có điều kiện cho tập lệnh kiểm tra. |
| Có thể export test sang định dạng được sử dụng trong Selenium RC và WebDriver. | Thực hiện kiểm thử chậm so với Selenium RC và WebDriver. |
| Được xây dựng để có thể report kết quả. |  |
| Cung cấp hỗ trợ cho các phần mở rộng. |  |

* 1. **Selenium WebDriver**
* Sự ra đời của WebDriver
  + Simon Stewart đã tạo ra WebDriver vào khoảng năm 2006, khi các trình duyệt và ứng dụng web đang trở nên mạnh mẽ hơn và hạn chế hơn với các chương trình JavaScript như Selenium Core. Đây là framework kiểm thử đa nền tảng đầu tiên có thể điều khiển trình duyệt từ cấp độ hệ điều hành.
* Sự ra đời của Selenium 2
  + Vào năm 2008, toàn bộ Nhóm Selenium quyết định kết hợp WebDriver và Selenium RC để tạo ra một công cụ mạnh mẽ hơn được gọi là Selenium 2, với WebDriver là trung tâm. Hiện tại, Selenium RC vẫn đang được phát triển nhưng chỉ ở chế độ bảo trì. Hầu hết những nỗ lực của Dự án Selenium hiện nay đang tập trung vào Selenium 2.
* Các ngôn ngữ được hỗ trợ giống như Selenium RC: Java, C#, PHP, Python, Ruby, Perl.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| Cài đặt đơn giản hơn so với Selenium RC | Cài đặt phức tạp hơn so với Selenium IDE |
| Giao tiếp trực tiếp với trình duyệt | Yêu cầu kiến thức về lập trình |
| Tương tác với trình duyệt có tính thực tế hơn | Không thể sẵn sàng hỗ trợ các trình duyệt mới |
| Không cần một thành phần riêng biệt như RC Server | Không có cơ chế tích hợp sẵn để ghi nhật ký thông điệp thời gian chạy và tạo kết quả kiểm thử |
| Thời gian thực thi nhanh hơn so với IDE và RC |  |

1. **Mục đích của việc kiểm thử tự động**
   1. **Tiết kiệm thời gian và tài nguyên**
   2. **Tự động hóa việc kiểm thử lặp đi lặp lại**
   3. **Tự động hóa các trường hợp kiểm thử phức tạp**
2. **Ưu và nhược điểm của Selenium**
3. **Cấu trúc của Selenium**
4. **Tìm hiểu bộ công cụ của Selenium**
5. **Cài đặt Selenium**
6. **Demo**

# **Danh mục tài liệu tham khảo**

[**https://viblo.asia/p/selenium-testing-phan-1-6J3ZgE7x5mB**](https://viblo.asia/p/selenium-testing-phan-1-6J3ZgE7x5mB)

[**https://www.selenium.dev/documentation/legacy/selenium\_1/**](https://www.selenium.dev/documentation/legacy/selenium_1/)

[**https://mindmajix.com/selenium/what-is-the-use-of-selenium-remote-control**](https://mindmajix.com/selenium/what-is-the-use-of-selenium-remote-control)

[**https://www.selenium.dev/documentation/ide/**](https://www.selenium.dev/documentation/ide/)

[**https://www.guru99.com/introduction-to-selenium.html**](https://www.guru99.com/introduction-to-selenium.html)

# **Chú thích:**

**[1] Same origin policy:** Chính sách cùng nguồn gốc là một cơ chế an ninh trong trình duyệt web để ngăn chặn các trang web hoặc tài nguyên từ nguồn không cùng nguồn gốc truy cập và tương tác với các dữ liệu từ nguồn khác. Nguồn gốc ở đây thường được xác định bằng cách so sánh giao thức, tên miền và cổng của hai trang web.

Ví dụ: Ví dụ: Giả sử bạn đang duyệt một trang web A từ miền "https://www.example.com". Trang web A muốn tải dữ liệu từ một trang web B có nguồn gốc từ "https://api.example.com". Nếu trang web B không chia sẻ cùng nguồn gốc với trang web A, thì trình duyệt sẽ áp dụng chính sách cùng nguồn gốc và ngăn chặn trang web A truy cập dữ liệu từ trang web B. Điều này là để đảm bảo rằng trang web A không thể lợi dụng quyền truy cập vào dữ liệu nhạy cảm trên trang web B mà người dùng không được phép truy cập.

Trong ngữ cảnh của Selenium, vấn đề xuất phát khi người kiểm thử cố gắng kiểm tra ứng dụng web trên một trang web khác nguồn gốc. Ví dụ, nếu bạn đang sử dụng Selenium để kiểm tra một trang web từ miền "https://www.example.com", nhưng Selenium được thực thi từ một miền khác, chẳng hạn "https://test.example". Chính sách cùng nguồn gốc sẽ ngăn chặn Selenium truy cập và tương tác với các thành phần của trang web từ miền "https://www.example.com".