# CÔNG CỤ KIỂM THỬ SELENIUM WEBDRIVER

## **Giới thiệu về công cụ Selenium**

### **Giới thiệu chung về Selenium**

Selenium (thường được viết tắt là SE) là một phần mềm mã nguồn mở, được phát triển bởi Jason Huggins, sau đó được tiếp tục phát triển bởi nhóm ThoughtWorks vào năm 2004. Selenium là một công cụ hỗ trợ kiểm tra tự động cho các ứng dụng chạy trên nền web.

Selenium hỗ trợ kiểm tra hầu hết trên các trình duyệt phổ biến hiện nay như Firefox, Internet Explorer, Chrome, Safari, cũng như các hệ điều hành chủ yếu như Windows, Linux, Mac... Selenium hỗ trợ một số lớn các ngôn ngữ lập trình như C#, Java, Perl, PHP, Python, Ruby... Không những vậy, Selenium có thể kết hợp thêm với một số công cụ khác như Bromien, Junit, Nunit.

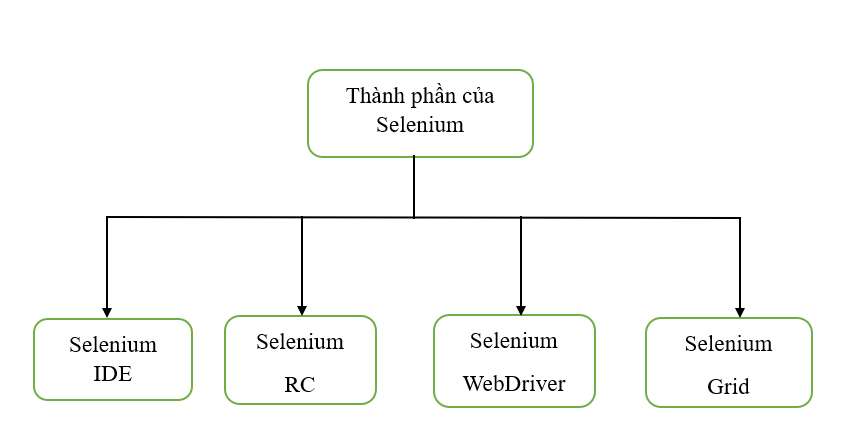
### **Đặc điểm của Selenium**

**Selenium có các đặc điểm sau:**

* Công cụ mã nguồn mở: Selenium là một công cụ mã nguồn mở, cho phép người dùng sử dụng và tùy chỉnh nó miễn phí.
* Hỗ trợ nhiều ngôn ngữ lập trình: Selenium hỗ trợ nhiều ngôn ngữ lập trình như Java, Python, C#, Ruby và JavaScript, giúp cho việc viết kịch bản kiểm thử linh hoạt và dễ dàng.
* Kiểm thử đa trình duyệt: Selenium có khả năng kiểm thử trên nhiều trình duyệt web phổ biến như Chrome, Firefox, Internet Explorer và Safari, đảm bảo tính tương thích của ứng dụng trên các nền tảng khác nhau.
* Các tính năng mạnh mẽ: Selenium cung cấp các tính năng mạnh mẽ như xác thực, ghi log, và kiểm tra hiệu suất, giúp người dùng kiểm tra và đảm bảo chất lượng của ứng dụng web một cách toàn diện.
* Cộng đồng sử dụng lớn: Selenium có một cộng đồng sử dụng rộng lớn, với nhiều tài liệu hướng dẫn, diễn đàn trao đổi thông tin, và các nguồn tài nguyên hữu ích khác, giúp người dùng dễ dàng học và sử dụng công cụ này.

### **Thành phần của Selenium**

Selenium gồm một bộ công cụ hỗ trợ kiểm tra tự động các ứng dụng website để đảm bảo chất lượng của các ứng dụng web, Selenium gồm có các thành phần sau:

****

*Hình 2. 1: Thành phần của Selenium*

* Selenium-IDE: Là một công cụ cho phép ghi (record) và phát lại (playback) một test script. Nó được thực hiện như là một Firefox Add-On. Selenium IDE cho phép xuất ra kịch bản đã thu dưới nhiều loại ngôn ngữ lập trình khác nhau như Java, Php, C#, Ruby, Perl hay Python.
* Selenium RC (Selenium Remote Control): Là một framework kiểm thử cho phép thực hiện nhiều hơn và tuyến tính các hành động trên trình duyệt. Nó cho phép cho phép các nhà phát triển tự động hóa kiểm thử sử dụng một ngôn ngữ lập trình cho tính linh hoạt tối đa và mở rộng trong việc phát triển logic thử nghiệm.
* Selenium WebDriver: Là sự kế thừa từ Selenium Remote Control, làm việc trực tiếp với trình duyệt ở mức hệ điều hành, cho phép gửi lệnh trực tiếp đến trình duyệt và xuất ra kết quả.
* Selenium-Grid: Là một hệ thống hỗ trợ người dùng thực thi test script trên nhiều trình duyệt một cách song song mà không cần phải chỉnh sửa test script. Thực hiện phương pháp kiểm tra phân bổ, phối hợp nhiều Selenium RC để có thể thực thi trên nhiều trình duyệt Web khác nhau trong cùng một lúc nhằm giảm thiểu thời gian thực hiện.

## **Công cụ kiểm thử tự động Selenium Webdriver**

### **Tổng quan về Selenium Webdriver**

WebDriver là một trong những công cụ mạnh mẽ và phổ biến nhất của bộ công cụ Selenium. WebDriver là phiên bản mở rộng của Selenium RC với những ưu điểm vượt trội và giải quyết được nhiều hạn chế của Selenium RC.

WebDriver mở rộng hỗ trợ cho nhiều trình duyệt và nền tảng mới nhất, không giống như Selenium IDE. WebDriver cũng không yêu cầu máy chủ Selenium phải được khởi động trước khi thực thi các tập lệnh thử nghiệm, không giống như Selenium RC.

### **Kiến trúc của Webdriver**

WebDriver là một công cụ kiểm thử dựa trên web với sự khác biệt tinh tế so với Selenium RC: WebDriver được xây dựng trên nền tảng cơ bản của một máy khách (client) độc lập được tạo cho mỗi trình duyệt web; không yêu cầu JavaScript nặng.

WebDriver thực hiện các cuộc gọi trực tiếp đến trình duyệt Web và toàn bộ tập lệnh kiểm thử được thực hiện theo kiểu này. WebDriver sử dụng được với các trình duyệt có hỗ trợ và có khả năng tự động hóa (automation).

Khác với Selenium RC, Selenium WebDriver về cơ bản không yêu cầu Selenium Server phải được khởi động trước khi khởi chạy thực thi tập lệnh kiểm thử. Người dùng có thể tận dụng tiện ích này để có thể yêu cầu hoặc không Selenium Server khi họ đang thực hiện kiểm thử trên cùng một máy nơi trình duyệt đang cư trú (cùng 1 client).

WebDriver là một framework hướng đối tượng hoàn toàn, hoạt động trên tầng OS (tầng hệ điều hành). Nó sử dụng khả năng tương thích tự nhiên của trình duyệt để tự động hóa mà không cần sử dụng bất kỳ thực thể ngoại vi nào. Với nhu cầu ngày càng tăng, WebDriver đã trở nên rất phổ biến, cơ sở người dùng lớn và cho đến nay đã trở thành một trong những công cụ kiểm thử tự động hóa nguồn mở được sử dụng rộng rãi nhất.

### **Các tính năng của Selenium Webdriver**

* **Tính tương thích của trình duyệt web (Browser Compatibility)**

Selenium WebDriver được thiết kế để hỗ trợ kiểm thử trên nhiều trình duyệt web phổ biến như Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge, và cả trình duyệt Internet Explorer (dù Selenium đã chuyển sang sử dụng Edge Chromium từ phiên bản 3.141.59).

Điều này đảm bảo rằng các kịch bản kiểm thử có thể chạy trên các trình duyệt khác nhau một cách đồng nhất và đáng tin cậy, giúp đảm bảo tính tương thích của ứng dụng web trên các nền tảng khác nhau.

* **Ngôn ngữ hỗ trợ (Language Support)**

Selenium WebDriver cung cấp khả năng viết kịch bản kiểm thử bằng một loạt các ngôn ngữ lập trình phổ biến. Các ngôn ngữ lập trình được Selenium WebDriver hỗ trợ:

* Java
* C#
* PHP
* Pearl
* Ruby
* Python

Điều này tạo điều kiện linh hoạt cho các nhà phát triển và kiểm thử viên lựa chọn ngôn ngữ phù hợp với nhu cầu cụ thể của dự án hoặc tổ chức.

* **Tốc độ nhanh (Speed)**

Selenium WebDriver được biết đến với hiệu suất cao và tốc độ kiểm thử nhanh chóng. Công cụ này giúp giảm thiểu thời gian thực hiện các thử nghiệm kiểm tra, từ việc mở trình duyệt đến thực hiện các thao tác trên các yếu tố trên trang web, tăng cường hiệu suất và hiệu quả của quá trình kiểm thử.

* **Cung cấp các trình điều khiển web (Drivers), phương thức (Methods) và lớp (Classes)**

Selenium WebDriver cung cấp một loạt các driver (như ChromeDriver, GeckoDriver, và WebDriver for Safari) cùng với phương thức và lớp lập trình cho việc kiểm thử.

Những công cụ này cho phép người dùng tương tác và kiểm soát các yếu tố trên trang web một cách dễ dàng và linh hoạt, từ việc tìm kiếm phần tử đến thực hiện các thao tác như nhấp chuột và nhập liệu. Điều này tạo ra một nền tảng mạnh mẽ cho việc tự động hóa kiểm thử và kiểm soát chất lượng phần mềm.

### **Ưu điểm và nhược điểm của Selenium Webdriver**

* **Ưu điểm**

- Mã nguồn mở: Selenium WebDriver là một phần mềm miền phí, dễ sử dụng và mang lại lợi ích trong việc kiểm thử phần mềm tự động.

- Tương thích nhiều trình duyệt: Selenium WebDriver tương thích với gần như tất cả các trình duyệt web phổ biến hiện nay, đảm bảo khả năng kiểm thử đa nền tảng.

- Hỗ trợ đa ngôn ngữ lập trình: Selenium WebDriver hỗ trợ gần hết các ngôn ngữ lập trình phổ biến hiện nay (C#, Java, Python, Ruby, Perl,…).

- Thân thiện, dễ sử dụng: Xét về độ thân thiện với người dùng, Selenium WebDriver vượt trội so với các thành phần khác của Selenium. Những câu lệnh của nó khá dễ nhớ và tiện dụng. Selenium cũng cung cấp cho người dùng nhiều tài liệu hướng dẫn cần thiết.

- Tốc độ thực thi nhanh: Selenium WebDriver tận dụng khá tốt khả năng hỗ trợ tự động hóa của các trình duyệt web. Tốc độ thực thi của Selenium WebDriver sẽ nhanh hơn nhiều nếu so với các công cụ khác trong bộ Selenium.

* **Nhược điểm**

- Chỉ hỗ trợ ứng dụng web: Selenium WebDriver chỉ hỗ trợ kiểm thử trên các ứng dụng web, hạn chế sự đa dạng trong kiểm thử ứng dụng.

- Đòi hỏi kinh nghiệm lập trình: Sử dụng Selenium WebDriver đòi hỏi tester phải có kiến thức vững về một ngôn ngữ lập trình cụ thể và kinh nghiệm sử dụng các công cụ kiểm thử tự động.

- Thiếu tính năng báo cáo tự động: Là cung cụ kiểm thử nhưng Selenium Webdriver chưa có sẵn chức năng báo cáo tự động nên việc khắc phục sự cố và sửa lỗi khó khăn hơn. Điều này đồng nghĩa với việc tester sẽ mất nhiều thời gian hơn để viết báo cáo kiểm thử.

## **Thư viện hỗ trợ Selenium Webdriver-TestNG**

Một trong những thư viện (framework) phổ biến nhất được sử dụng với Selenium WebDriver là TestNG.

TestNG là một framework kiểm thử mạnh mẽ cho Java, được thiết kế để cải thiện khả năng quản lý và thực thi kiểm thử, đồng thời cung cấp nhiều tính năng linh hoạt và tiện ích bao gồm:

* *Annotations:* thư viện TestNG cung cấp các annotation như @Test, @BeforeSuite, @AfterSuite để đánh dấu các phương thức là các bước kiểm thử, trước hoặc sau toàn bộ bộ kiểm thử hoặc nhóm kiểm thử.
* Bộ kiểm thử linh hoạt: TestNG cho phép tổ chức các phương thức kiểm thử thành các nhóm và phân loại chúng bằng cách sử dụng các annotation.
* *Quản lý thời gian:* Bạn có thể cấu hình TestNG để thực hiện các bước kiểm thử theo thứ tự nhất định hoặc song song, cũng như đặt thời gian timeout cho mỗi kiểm thử.
* *Báo cáo và ghi nhật ký:* TestNG tạo ra báo cáo chi tiết về kết quả kiểm thử, bao gồm thông tin về số lượng kiểm thử thành công và thất bại, thời gian chạy, và các chi tiết về lỗi nếu có.

Nhờ những tính năng linh hoạt và tiện ích của nó, TestNG là một trong những lựa chọn phổ biến cho việc triển khai kiểm thử tự động với Selenium WebDriver trong cộng đồng phát triển phần mềm Java.

Cài đặt Selenium WebDriver và TestNG với ngôn ngữ Java

* Các bước cài đặt Selenium WebDriver:

Bước 1: Cài đặt Chrome Brower

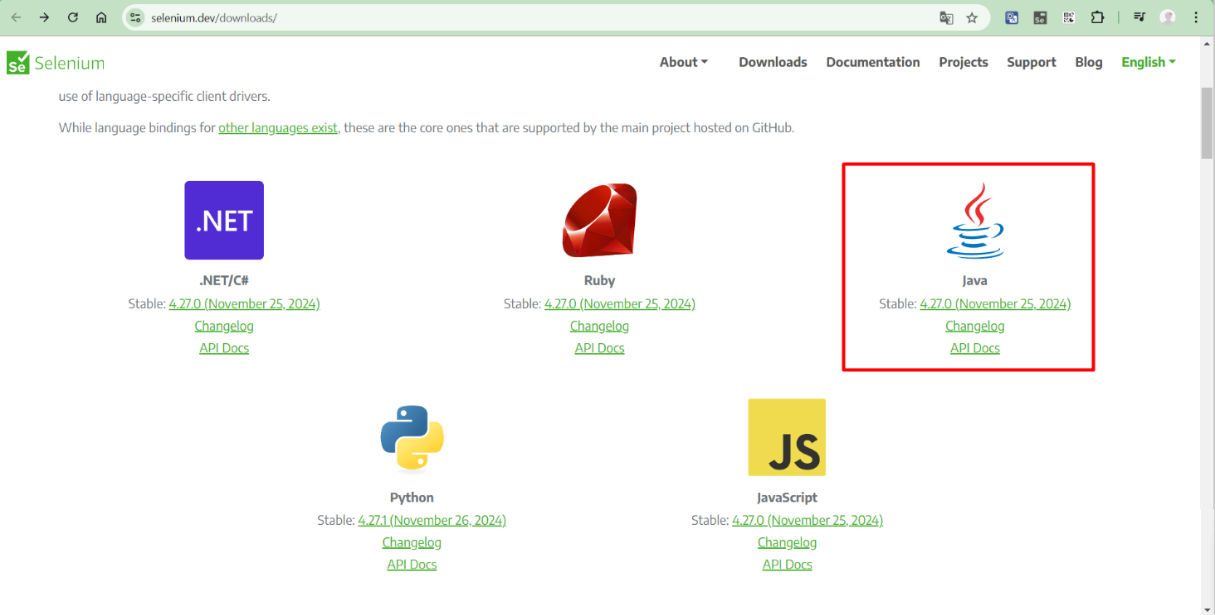
* + Vào trang web <https://www.google.com/intl/vi_vn/chrome/> và nhấn nút “Tải xuống tại đây”.
  + Thực hiện cài đặt Chrome.

Bước 2: Cài đặt Chrome Driver

* + Vào trang web <https://chromedriver.chromium.org/downloads> và tải phiên bản phù hợp với Chrome Browser đã tải ở bước 1.

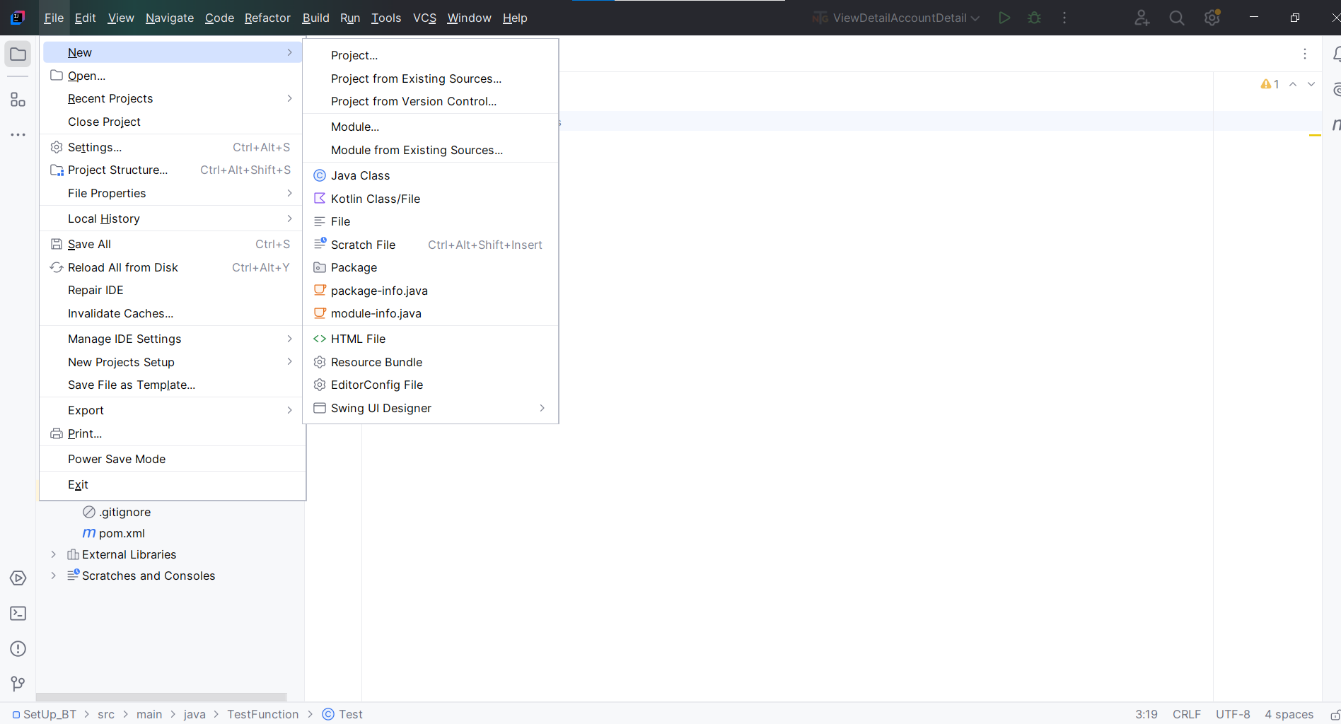
Bước 3: Cài đặt thư viện Selenium Webdriver sử dụng ngôn ngữ Java vào IntelliJ IDEA.

Trước khi cài đặt ta phải có điều kiện sử dụng Java 8 trở lên và có công cụ hỗ trợ code (Intellij IDEA, Eclipse,..). Ở đây ta thực hiện cài đặt Selenium WebDriver với công cụ IntelliJ IDEA.



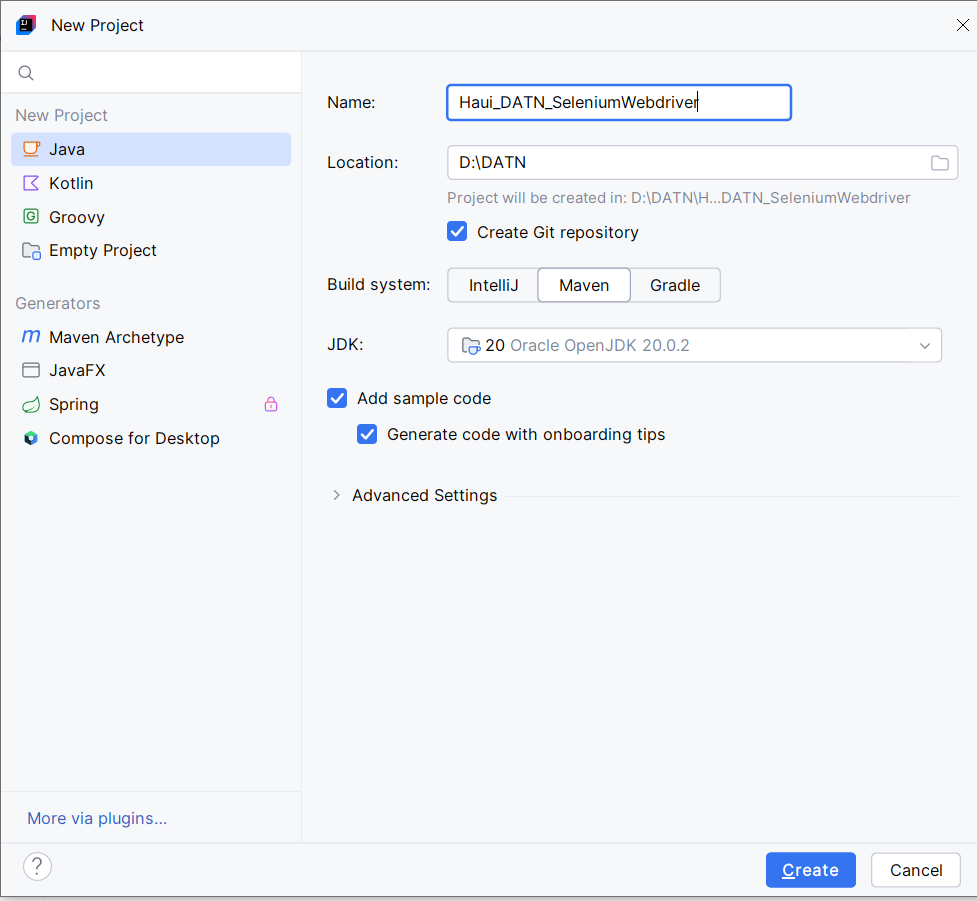
*Hình 2. 2: Màn hình trang tải về Selenium Webdriver*

Mở IntelliJ IDEA lên, tạo một project mới.

****

*Hình 2. 3: Thao tác tạo một project mới trong IntelliJ*

Chọn cấu hình JDK đúng với version Java.



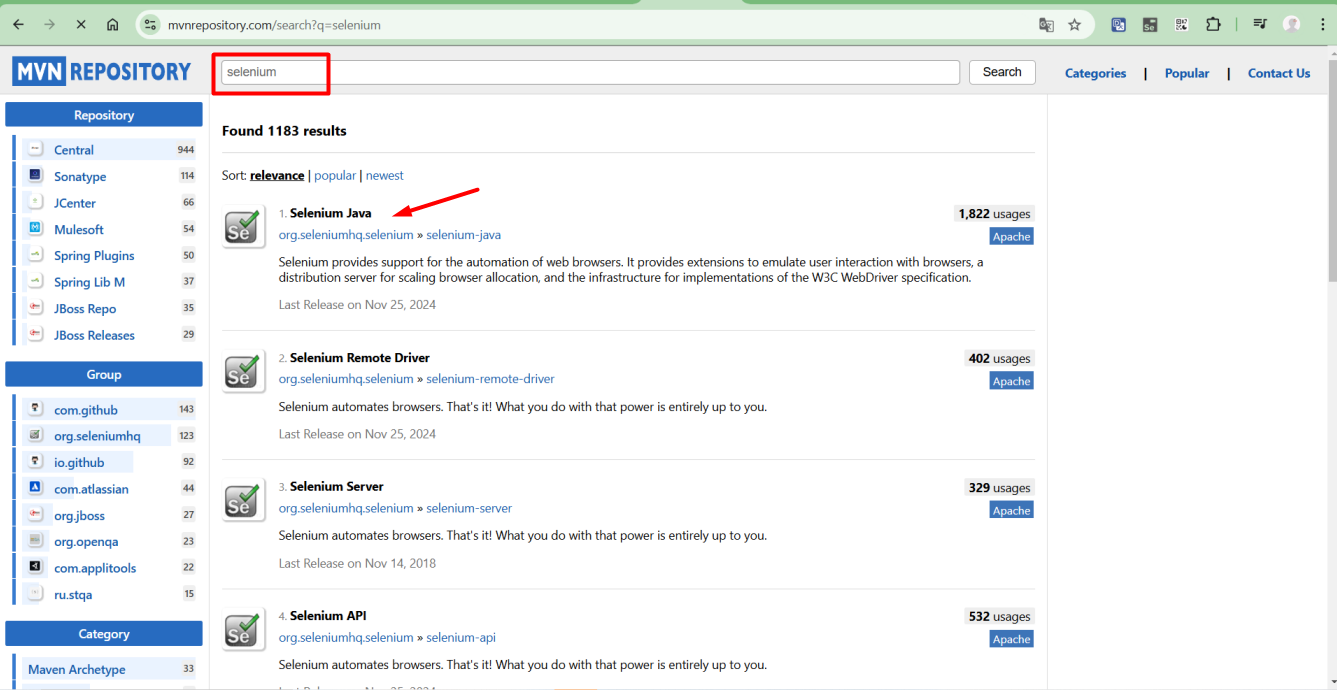
*Hình 2. 4: Chọn cấu hình JDK đúng với version Java ở IntelliJ*

Sau khi tạo Project Java xong, tới một bước quan trọng là cài đặt thư viện của Selenium WebDriver vào project để nó có thể chạy được các yêu cầu của WebDriver.

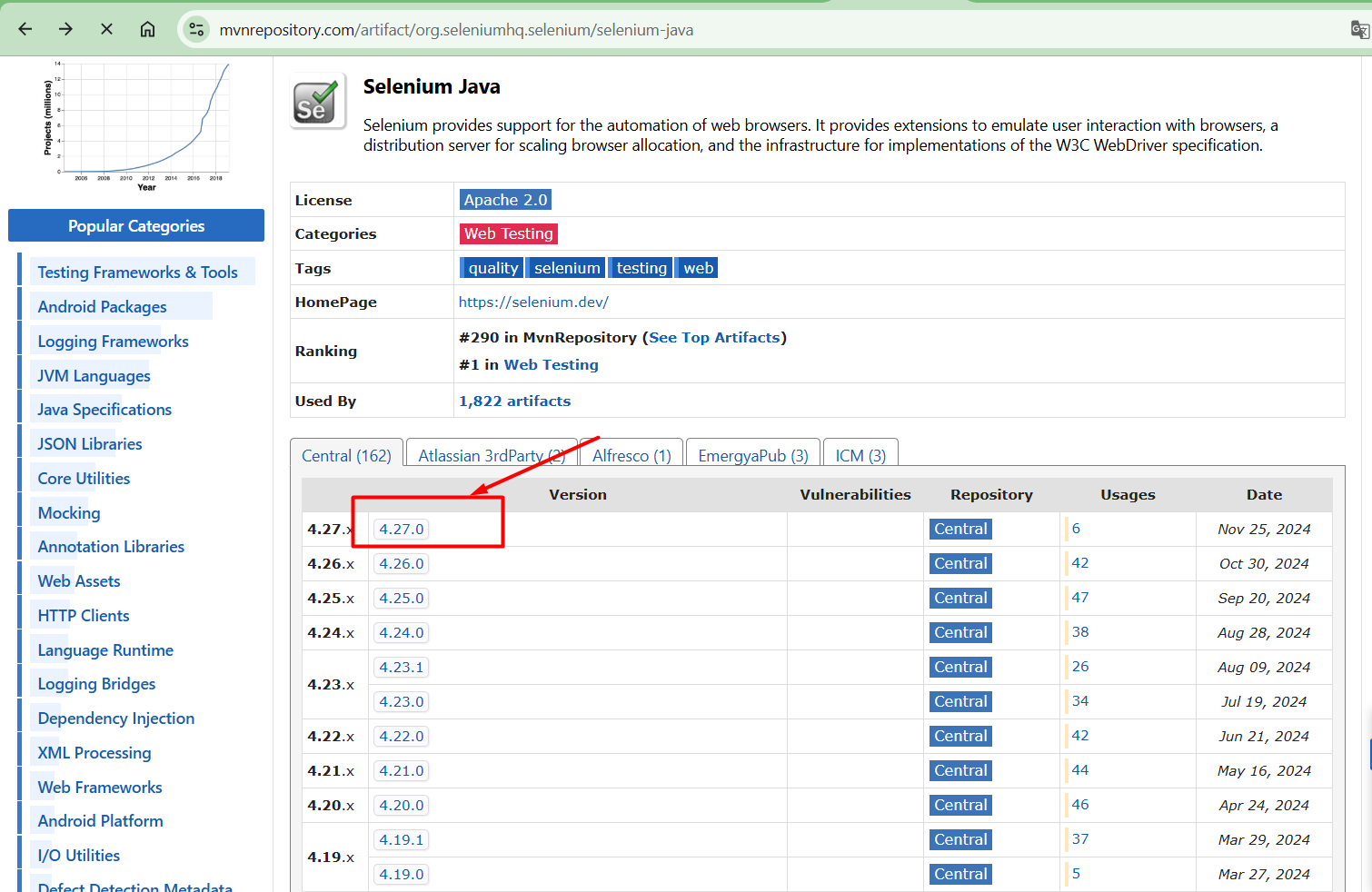
* Cài đặt thư viện Selenium vào Maven project trên IntelliJ

Để cài đặt thư viện Selenium vào Maven project thì sẽ cài nó vào file pom.xml

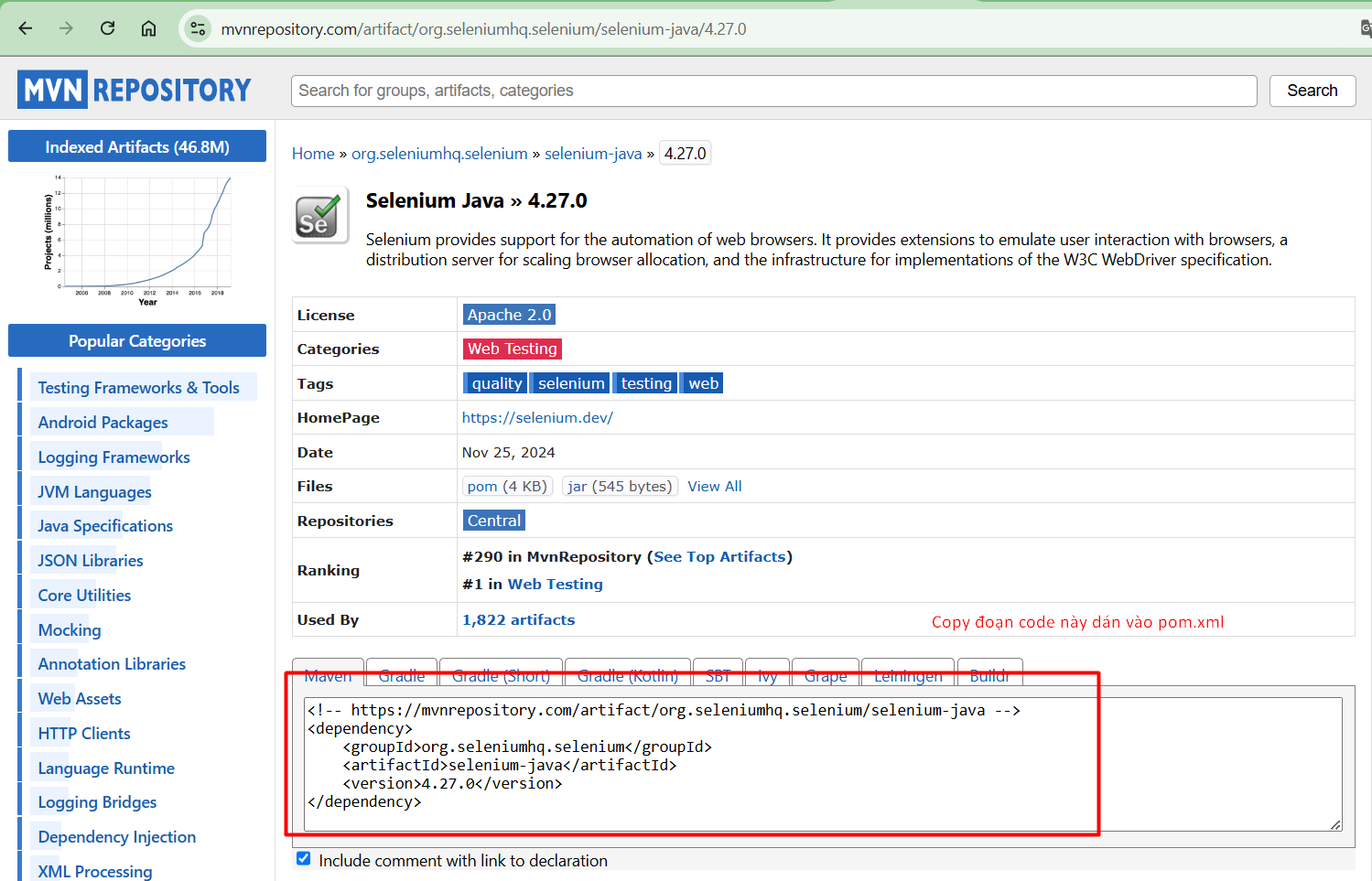
Thư viện được lấy tại trang web: <https://mvnrepository.com/>

****

*Hình 2. 5: Thao tác tải Selenium Java*

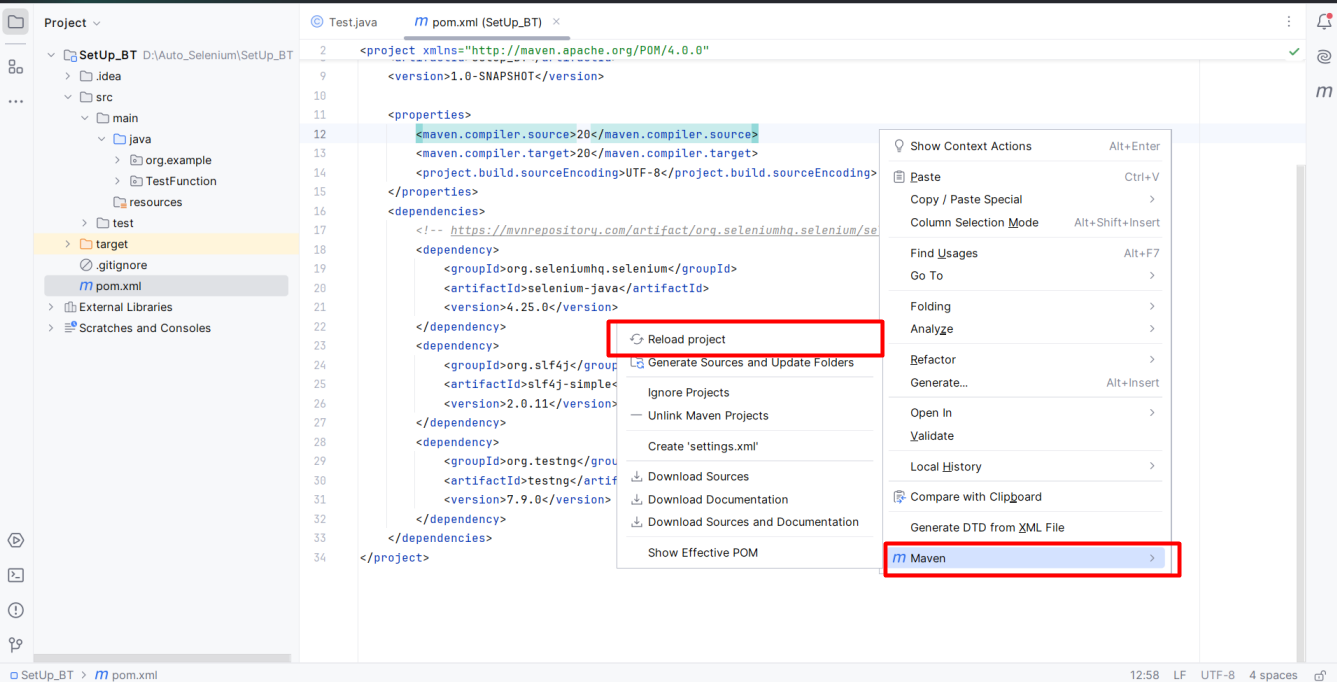
****

*Hình 2. 6: Chọn phiên bản phù hợp*

****

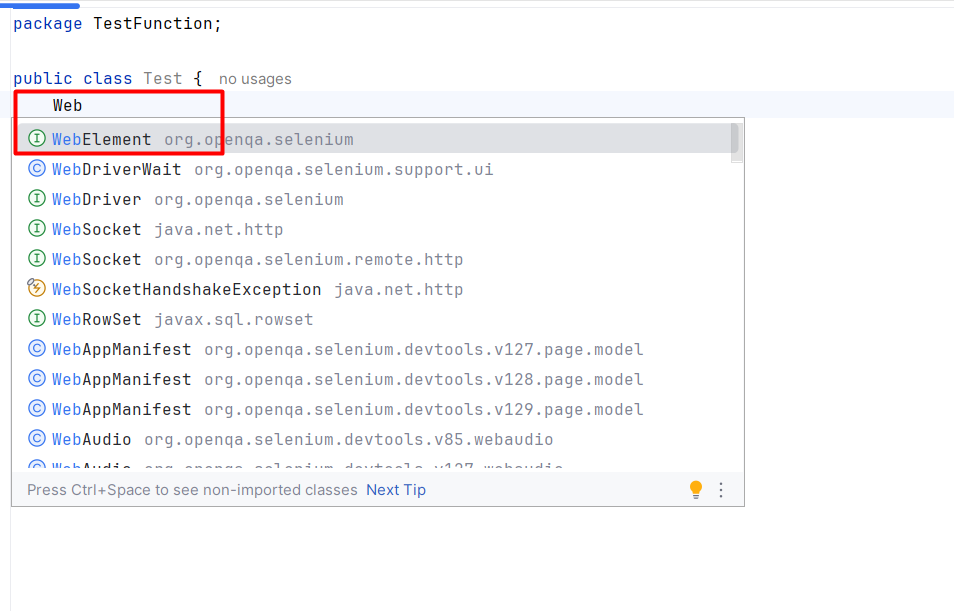
*Hình 2. 7: Thao tác cài đặt Selenium Java về project*

Khi dán vào file pom.xml nhớ để trong cặp thẻ <dependencies> </dependencies>. Sau khi dán xong nhớ reload project lại bằng cách nhấn vào biểu tượng Reload hoặc chuột phải > Maven > reload project.



*Hình 2. 8: Thực hiện khởi động lại Maven*

Kiểm tra Selenium đã cài đặt vào maven project

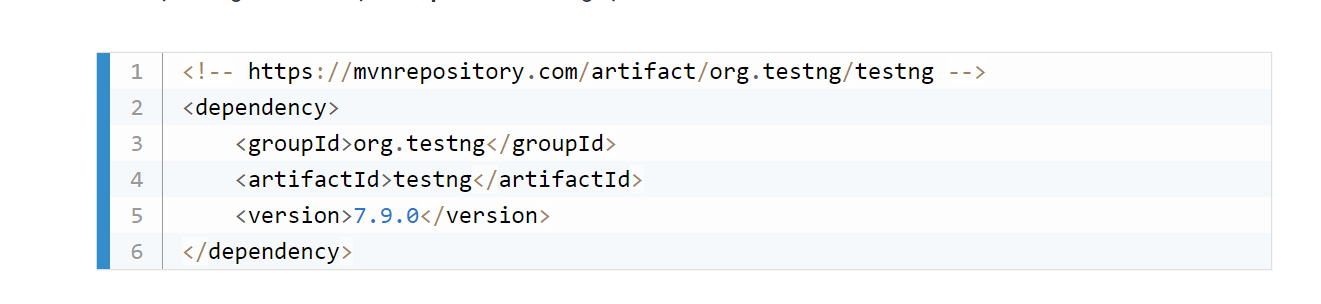
****

*Hình 2. 9: Hoàn thiện cài đặt thư viện Selenium*

Nếu gợi ý hiển thị lên như hình là đã cài đặt được Selenium thành công

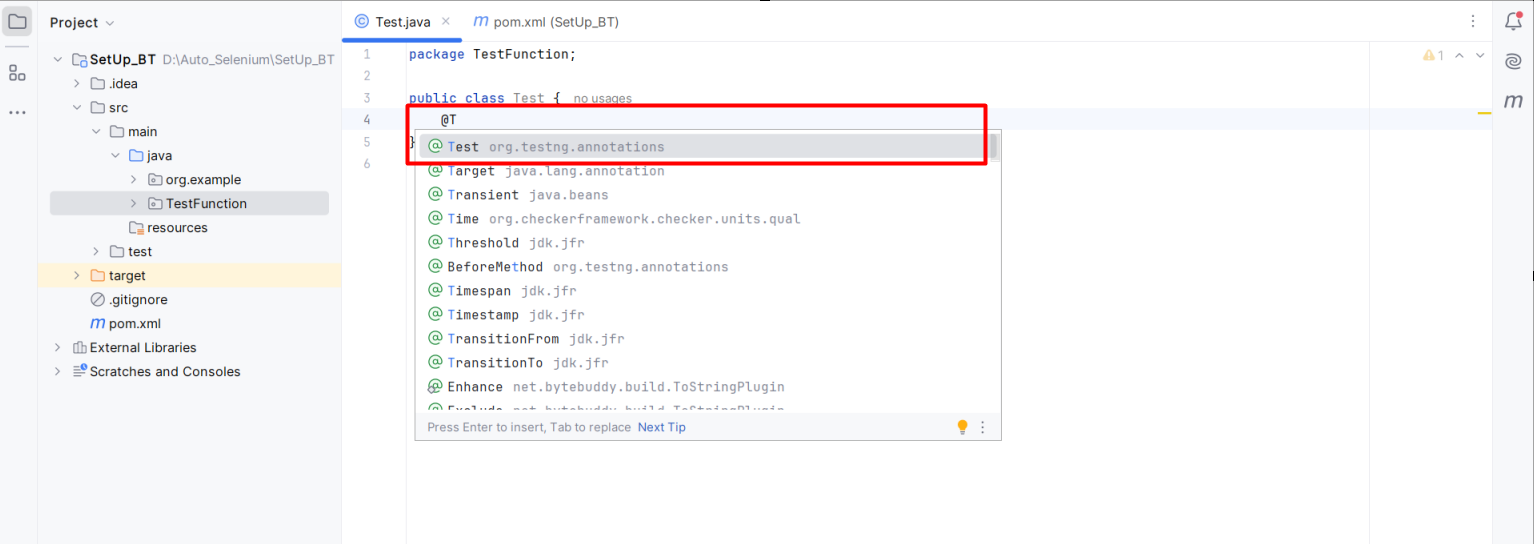
Bước 4: Cài đặt thư viện TestNG vào IntelliJ

Thêm thư viện vào file pom.xml tương tự Selenium



*Hình 2. 10: Cài đặt thư viện TestNG*

Thực hiện Reload Maven để hoàn tất cài đặt

****

*Hình 2. 11: Hoàn thiện cài đặt TestNG*

## **Kết luận**

Trong chương 2 của đồ án đã đưa ra lý thuyết cơ bản về Selenium và đặt Selenium Webdriver:

* Giới thiệu về công cụ Selenium.
* Công cụ kiểm thử tự động Selenium Webdriver.
* Thư viện hỗ trợ TestNG.
* Cài đặt Selenium Webdriver và TestNG với ngôn ngữ Java vào Intellij.

Trong chương 3, em sẽ tìm hiểu về website Nhà Sách Phương Nam, trình bày kịch bản kiểm thử và ứng dụng trực tiếp Selenium WebDriver để kiểm chức năng của website Nhà Sách Phương Nam