

	VIETTEL AI RACE	TD179
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG (AUTOMATION TESTER)	Lần ban hành: 1

1. Kiểm thử tự động là gì?

Định nghĩa: Kiểm thử tự động là một kỹ thuật kiểm thử phần mềm thực hiện bằng cách sử dụng các công cụ kiểm thử tự động đặc biệt để thực hiện một bộ trường hợp kiểm thử. Về cơ bản, đây là một bài kiểm tra để kiểm tra lại xem thiết bị hoặc phần mềm có thực hiện chính xác như theo thiết kế hay không. Trong quá trình phát triển sản phẩm, kiểm thử tự động sẽ kiểm tra để tìm ra các lỗi, các sai sót hay bất kỳ vấn đề phát sinh nào khác. Kiểm thử tự động được sử dụng cho các công việc lặp đi lặp lại cũng như các công việc khó có thể thực hiện bằng kiểm thử thủ công.

2. Tầm quan trọng của kiểm thử tự động

- Ngày nay có rất nhiều dự án/doanh nghiệp không có đủ thời gian và nguồn nhân lực cho việc kiểm thử thủ công. Tự động hóa trở thành lựa chọn hàng đầu của họ vì tính chính xác, nhanh chóng, hiệu quả và không tốn nhiều nguồn lực. Điều này thực sự thúc đẩy năng suất của các dự án/doanh nghiệp.
- Các chu kỳ phát triển phần mềm luôn yêu cầu việc kiểm tra lặp đi lặp lại. Kiểm thử tự động sẽ giúp cho việc này trở nên khả thi mà không làm mất đi nguồn lực của các thành viên trong nhóm dự án/doanh nghiệp. Nó cũng có thể mang lại kết quả chính xác và đáng tin cậy hơn so với việc chỉ kiểm tra thủ công. Ngoài ra sản phẩm còn được đảm bảo để sẵn sàng đưa ra thị trường hoặc chuyển sang giai đoạn phát triển tiếp theo. Việc này mang lại cho nhóm dự án động lực để tiếp tục phát triển sản phẩm của mình.
- Quan trọng hơn cả, kiểm thử tự động giúp ích cho việc phát triển sản phẩm bởi khi ứng dụng kiểm thử tự động vào chu trình sản xuất, nó sẽ tạo tiền đề cho sự phát triển liên tục sau khi sản phẩm ra mắt. Nhờ kiểm thử tự động doanh nghiệp/nhóm dự án có thể làm việc trên nhiều phần mềm và sản phẩm hơn, ngay cả khi lượng thành viên trong nhóm không thay đổi. Điều này có nghĩa là họ không chỉ hoàn thiện sản phẩm cuối, mà họ còn luôn tạo ra thêm những phần mềm/sản phẩm mới.

3. Quy trình kiểm kiểm thử tự động

Một quy trình kiểm thử tự động thường trải qua những bước cơ bản như sau:

B1: Lựa chọn công cụ kiểm thử tự động

B2: Xác định phạm vi

B3: Lập kế hoạch

B4: Thiết kế kịch bản kiểm thử tự động

B5: Phát triển kịch bản kiểm thử tự động

	VIETTEL AI RACE	TD179
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG (AUTOMATION TESTER)	Lần ban hành: 1

B6: Thực thi kiểm thử (Execution)

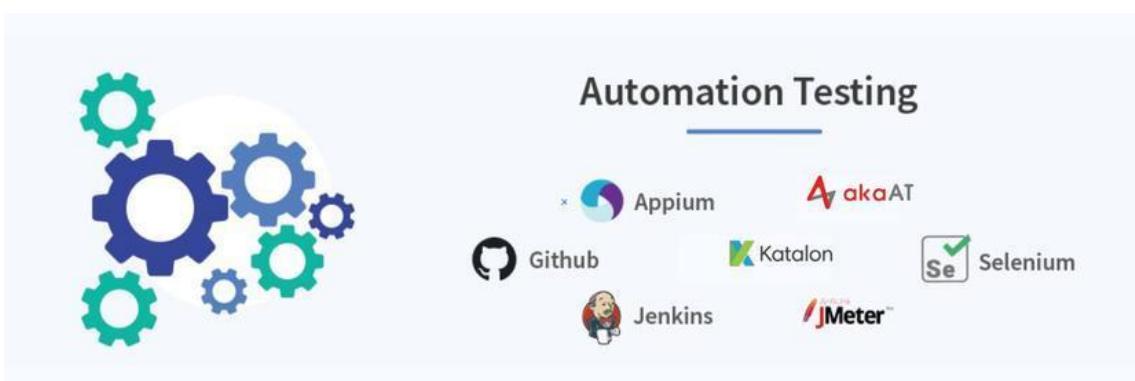
B7: Cập nhật và Bảo trì

3.1 Bước 1: Lựa chọn công cụ kiểm thử

Việc lựa chọn công cụ kiểm thử tự động phù hợp thường phụ thuộc vào Loại của Ứng dụng đang được kiểm tra (Application Under Test) và Môi trường mà ứng dụng đó sẽ chạy.

Ví dụ: Nếu ứng dụng của bạn là ứng dụng Web chạy trên Web browser của máy tính cá nhân (Desktop), thì Selenium Webdriver có thể là lựa chọn phù hợp. Nhưng nếu ứng dụng Web của bạn có cả phiên bản chạy được trên browser của các thiết bị Mobile, thì có thể bạn sẽ cần thêm 1 công cụ có thể hỗ trợ tạo kịch bản test chạy được cho cả các Browsers trên Mobile nữa. Khi đó Appium có thể là một lựa chọn phù hợp. Hiện tại trên thị trường có đa dạng các loại công cụ kiểm thử tự động, từ miễn phí (Open Source, Freeware), đến có phí (License). Chẳng hạn Opensource có thể kể đến như: Selenium IDE, Selenium WebDriver, Appium, Cucumber, hay có License như: Postman, Katalon, akaAT, Ranorex, Tosca Tricentis, Testcomplete, HP-UFT. Các công cụ mã nguồn mở thường yêu cầu bạn phải có kỹ năng làm việc với ngôn ngữ lập trình tương đối thuần thục để có thể đọc hiểu được framework và viết kịch bản dưới dạng test script để tận dụng được sức mạnh của các chức năng mà thư viện công cụ đó cung cấp. Vì thế cũng tốn nhiều thời gian và công sức hơn mới tạo được các kịch bản như mong muốn. Đổi lại, nó cho bạn sự tự do nhất định khi được lựa chọn làm việc với ngôn ngữ lập trình mà bạn yêu thích. Ngược lại, các công cụ có trả phí (License) thường cung cấp giao diện trực quan cùng các tiện ích và nhiều tính năng hỗ trợ tối đa cho người dùng có thể tạo ra kịch bản nhanh chóng, đơn giản mà không phải code, chỉ phải kéo thả, Record và config. Nó giúp bạn tập trung vào việc quản lý Logic của ứng dụng và chất lượng của test case. Là công cụ có License nên bạn hoặc công ty sẽ phải trả phí hàng năm cho việc tiếp tục sử dụng công cụ và nhận được sự hỗ trợ từ nhà cung cấp công cụ đó. Vậy nên để lựa chọn đúng bạn cần tìm hiểu về các loại công cụ khác nhau và cách nó có thể mang lại lợi ích cho công việc kiểm thử của bạn.

	VIETTEL AI RACE	TD179
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG (AUTOMATION TESTER)	Lần ban hành: 1



3.2 Bước 2: Xác định phạm vi Kiểm thử tự động

Trong bước này, bạn sẽ xem xét khoanh vùng các luồng nghiệp vụ test phù hợp cho việc tự động hóa nó, chuẩn bị dữ liệu và môi trường diễn ra kiểm thử. Dưới đây là một số yếu tố cần xem xét khi xác định phạm vi thử nghiệm tự động hóa của bạn: Các tính năng chính, quan trọng của ứng dụng; Các trường hợp kiểm thử có nhiều dữ liệu; Các tính năng dùng chung (common) trên ứng dụng; Những vùng khả thi về kỹ thuật mà công cụ đáp ứng được; Những nghiệp vụ hay được tái sử dụng; Mức độ phức tạp của các test cases; Khả năng sử dụng các trường hợp kiểm thử giống nhau để test trên nhiều trình duyệt (với trường hợp kiểm thử ứng dụng web)

3.3 Bước 3: Lập kế hoạch

Trong giai đoạn này trước hết bạn sẽ xây dựng bản Kế hoạch bao gồm: Thông tin lộ trình thực hiện (Master Schedule); Effort và thời gian cho việc thiết kế Framework; Effort và thời gian cho việc viết kịch bản; Effort cho việc thực thi và báo cáo; Effort cho bảo trì sau phát triển; Các rủi ro phát sinh

3.4 Bước 4 + Bước 5: Thiết kế và Phát triển

Ở giai đoạn này, các thành viên của nhóm sẽ bắt tay vào thiết kế các kịch bản tự động và dùng công cụ được chọn để tạo ra các script tự động hóa. Các công việc điển hình như: Thiết kế Framework và các tính năng của nó; Thiết kế các kịch bản tự động; Viết script và kiểm tra tính ổn định của script; Review tính đúng đắn của script so với thiết kế

3.5 Bước 6: Thực thi kiểm thử (Execution)

Đây là giai đoạn áp chót trong quy trình kiểm thử tự động hóa. Khi bạn đã tạo xong các script tự động của mình, đã đến lúc chạy các script này để thực hiện kiểm thử ngay trên ứng dụng. Kết quả của việc thực thi các script thường sẽ được tổng hợp vào một báo cáo cho biết số lượng test cases PASS/FAIL và kèm theo các bằng chứng (Evidence) ghi lại hình ảnh/trạng thái tại thời điểm xảy ra các lỗi được tìm thấy.

3.6 Bước 7: Bảo trì

	VIETTEL AI RACE	TD179
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG (AUTOMATION TESTER)	Lần ban hành: 1

Khi phần mềm cần test (Application Under Test) được cập nhật thêm tính năng mới hoặc chỉnh sửa, thì bộ script sẽ được chạy để kiểm tra xem mức độ ảnh hưởng của các tính năng mới tới các tính năng hiện tại, liệu chúng có còn hoạt động đúng hay không. Và khi phần mềm có cập nhật, những thay đổi trên phần mềm có thể làm cho bộ script không còn đúng. Đó là lúc bạn cần có hoạt động xem xét chỉnh sửa, điều chỉnh lại script cho phù hợp với những thay đổi, cũng như thường xuyên nâng cấp nhằm nâng cao hiệu quả của các script tự động ở mỗi chu kỳ phát hành tiếp theo

4. Các loại kiểm thử tự động và cách apply vào dự án

Kiểm thử tự động giúp các nhóm dự án và tổ chức tự động kiểm thử phần mềm/sản phẩm của mình nhằm giảm thiểu tối đa sự can thiệp của con người, từ đó dự án đạt được tốc độ, độ chính xác và hiệu quả cao hơn. Chúng ta có thể phân các loại kiểm thử tự động như sau:

- Unit Testing
- Linear Testing
- Keyword Driven Testing
- Data Driven Testing
- Behavior Driven Testing
- Performance Testing

Để có thể lựa chọn được công cụ và loại hình kiểm thử tự động phù hợp, Bạn hãy tham khảo các gợi ý sau:

4.1 Tìm hiểu nhu cầu của dự án và nhu cầu kiểm thử

Việc cân nhắc liệu "kiểm thử tự động" có phải là hướng đi phù hợp cho nhóm dự án của bạn hay không là rất quan trọng. Vì không phải tất cả các nhóm QA đều cần tự động hóa để đẩy nhanh quá trình thử nghiệm của họ. Kiểm thử thủ công vẫn đóng 1 vai trò rất quan trọng trong quy trình sản xuất phần mềm/sản phẩm. Vì vậy, chỉ thực hiện kiểm thử tự động khi:

- Dự án có nhiều trường hợp kiểm thử lặp đi lặp lại cần thực hiện
- Khi có các bài kiểm tra hồi quy thường xuyên
- Khi nhóm dự án phải mô phỏng một số lượng lớn người dùng để kiểm tra hiệu suất
- Khi giao diện người dùng ổn định
- Khi các chức năng quan trọng không thể chỉ dựa vào thử nghiệm thủ công

4.2 Tiêu chí đánh giá công cụ kiểm thử tự động

Kiểm thử tự động mang tính kỹ thuật cao hơn nhiều so với kiểm thử thủ công. Với nhiều công cụ kiểm thử tự động, đặc biệt là phần mềm mã nguồn mở, người kiểm thử phải có đủ kiến thức lập trình để viết và thực thi các kịch bản kiểm thử. Rào cản kỹ thuật này dường như là thách thức lớn nhất trong việc áp dụng tự động hóa

	VIETTEL AI RACE	TD179
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG (AUTOMATION TESTER)	Lần ban hành: 1

thử nghiệm cho các nhóm QA có nền tảng CNTT hạn chế. Tuy nhiên, ngày nay các công cụ kiểm tra không yêu cầu code trong quá trình thực thi đã được chứng minh là một giải pháp đầy hứa hẹn cho vấn đề này.

4.2.1 Ngân sách nhóm dự án là bao nhiêu?

Thực tế chi phí để thực thi kiểm thử tự động khá cao. Nhưng đổi lại, nó sẽ mang lại tỷ suất hoàn vốn tích cực cho nhóm và doanh nghiệp về lâu dài nếu ngân sách được tính toán kỹ lưỡng. Khi nắm được ngân sách hoạt động, bạn sẽ dễ dàng hơn trong việc chọn phần mềm thích hợp với dự án của mình.

4.2.2 Cần tìm kiếm những tính năng gì?

Mặc dù yêu cầu của các nhóm dự án là khác nhau, nhưng vẫn có một số yếu tố chính mà bạn luôn phải cân nhắc khi chọn một công cụ kiểm thử tự động. Cụ thể:

- Nền tảng được hỗ trợ
- Ứng dụng được áp dụng theo thử nghiệm
- Ngôn ngữ lập trình
- Khả năng tích hợp CI/CD
- Chức năng báo cáo

4.2.3 Khả năng tích hợp như thế nào?

Công cụ kiểm thử tự động được chọn phải có khả năng tích hợp CI/CD và các nền tảng bên ngoài để đảm bảo tính liên tục của thử nghiệm. Tích hợp mạnh mẽ và toàn diện cũng cho phép bạn quản lý thử nghiệm và cộng tác nhóm tốt hơn.