

	VIETTEL AI RACE	TD067
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG VÀ CÔNG CỤ HỖ TRỢ	Lần ban hành: 1

Kiểm thử đang được xem là giải pháp chủ yếu nhằm đảm bảo chất lượng cho các sản phẩm phần mềm. Tuy nhiên, các hoạt động kiểm thử hiện nay chủ yếu được thực hiện một cách thủ công và tiêu tốn khoảng 30-50% tài nguyên (thời gian, nhân lực và chi phí) của quá trình phát triển sản phẩm phần mềm. Hơn nữa, độ phức tạp của các phần mềm ngày càng tăng và trong môi trường cạnh tranh như hiện nay đòi hỏi các công ty phần mềm phải áp dụng các phương pháp và công cụ nhằm tự động hóa các hoạt động kiểm thử. Hơn nữa, các công cụ kiểm thử tự động là một giải pháp hữu hiệu cho kiểm thử hồi quy. Chương này giới thiệu tổng quan về kiểm thử tự động và các công cụ hỗ trợ nhằm giải quyết vấn đề này.

1. Tổng quan về kiểm thử tự động

Một thực tế đáng buồn hiện nay là chất lượng của hầu hết các sản phẩm phần mềm rất thấp. Hàng năm, chúng ta phải chịu thiệt hại nhiều tỷ đô la do các lỗi phần mềm gây ra [oST02a, G.95]. Theo thống kê của NIST công bố năm 2002 [oST02b], chất lượng phần mềm thấp đã gây thiệt hại cho kinh tế Mỹ 60 tỷ đô la mỗi năm và tiêu tốn khoảng 22 tỷ đô la cho việc phát triển các công cụ nhằm phát hiện các lỗi và kiểm thử tự động. Có hai lý do chính dẫn đến tình trạng này. Thứ nhất, hầu hết các công cụ hiện nay đều tập trung vào việc thực thi tự động các ca kiểm thử (auto-test execution) trong khi vấn đề cốt lõi của kiểm thử là các phương pháp và kỹ thuật sinh các ca kiểm thử vẫn còn thiếu. Thứ hai, các công cụ hiện nay chưa hỗ trợ một cách hiệu quả cho kiểm thử hồi quy (regression testing). Một khi phần mềm bị tiến hóa/thay đổi, chúng ta cần kiểm thử lại sản phẩm. Làm thế nào để sử dụng lại các ca kiểm thử đã có và sinh ra các ca kiểm thử mới một cách hiệu quả đang là một vấn đề mở và chưa có giải pháp thỏa đáng. Hơn nữa, một trong những vấn đề khó nhất của kiểm thử tự động đó là việc sinh các giá trị đầu ra mong đợi tương ứng với các đầu vào của các ca kiểm thử. Đây là một bài toán khó và chưa có giải pháp hiệu quả nhằm giải quyết vấn đề này.

Giải pháp chủ yếu để giải quyết các vấn đề trên là đề xuất các phương pháp và công cụ hỗ trợ tối đa các hoạt động trong quy trình kiểm thử phần mềm. Trong quy trình kiểm thử, chúng ta cần một số công cụ ứng với các pha và các mục tiêu kiểm thử khác nhau. Ví dụ, để tự động hóa chiến lược kiểm thử hộp đen, phương pháp kiểm thử dựa trên mô hình (model-based testing) đang được biết đến như là một giải pháp tin cậy và hiệu quả [MRA04, KJ02, BFM04]. Với kiểm thử hộp trắng, ứng với mỗi phương pháp khác nhau chúng ta cũng có rất nhiều công cụ hỗ trợ. Ngoài ra, một số công cụ hỗ trợ kiểm thử các tính chất phi chức năng như độ an toàn, bảo mật, hiệu năng và khả năng chịu tải, v.v., cũng đã được phát triển và sử dụng rộng rãi.

Kiểm thử tự động là quá trình thực hiện một cách tự động các bước trong một kịch bản kiểm thử. Kiểm thử tự động bằng một công cụ nhằm rút ngắn thời gian kiểm thử. Mục đích của kiểm thử tự động là giảm thiểu thời gian, công sức và kinh phí trong khi vẫn tăng độ tin cậy, tăng tính hiệu quả và giảm sự nhảm chán cho người kiểm thử trong quá trình kiểm thử sản phẩm phần mềm. Kiểm thử tự động sẽ được

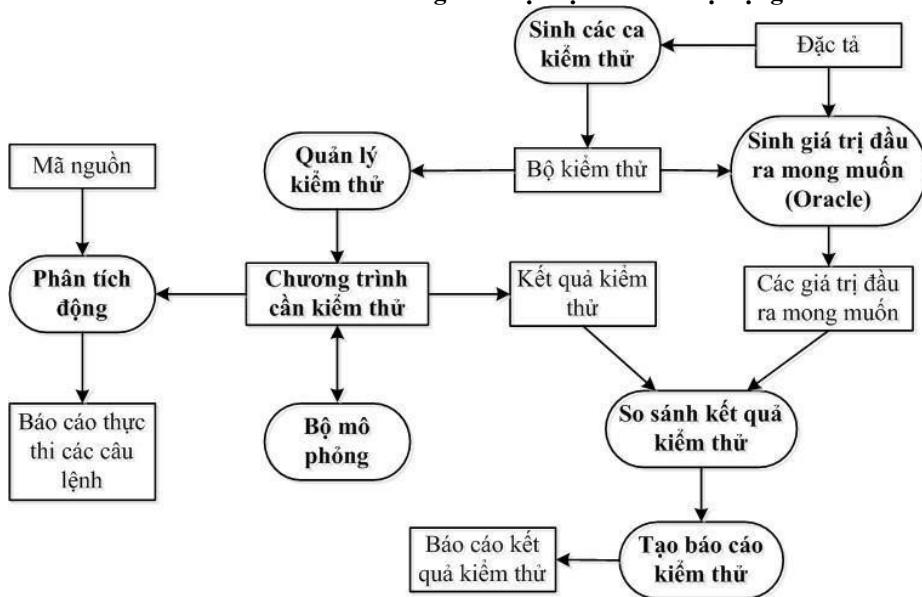
	VIETTEL AI RACE	TD067
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG VÀ CÔNG CỤ HỖ TRỢ	Lần ban hành: 1

sử dụng khi dự án không đủ tài nguyên (thời gian, nhân lực và chi phí), phải thực hiện kiểm thử hồi quy khi sản phẩm được sửa đổi hoặc nâng cấp và cần kiểm thử lại các tính năng đã thực hiện tốt trước đó, kiểm tra khả năng vận hành của sản phẩm trong các môi trường đặc biệt (độ tốc độ xử lý trung bình ứng với mỗi yêu cầu, xác định khả năng chịu tải tối đa, xác định cấu hình tối thiểu để thực thi hệ thống, kiểm tra các cơ chế an ninh và an toàn, v.v.).

2. Kiến trúc của một bộ công cụ kiểm thử tự động

Trong thực tế, có rất nhiều bộ công cụ hỗ trợ kiểm thử tự động được phát triển nhằm góp phần giải quyết các vấn đề khó khăn của quy trình kiểm thử. Hình 9.1 mô tả kiến trúc chung nhất của một bộ kiểm thử tự động [Som10]. Trong kiến trúc này, các công cụ kiểm thử được tích hợp trong một quy trình thống nhất nhằm hỗ trợ đầy đủ các hoạt động kiểm thử trong quy trình kiểm thử các sản phẩm phần mềm.

Hình 9.1: Kiến trúc chung của một bộ kiểm thử tự động.



Các công cụ cơ bản trong kiến trúc này bao gồm:

- **Quản lý kiểm thử:** công cụ này cho phép quản lý việc thực hiện/thực thi các ca kiểm thử. Nó giám sát việc thực hiện từng ca kiểm thử ứng với bộ giá trị đầu vào, giá trị đầu ra mong muốn và giá trị đầu ra thực tế. JUnit là một ví dụ điển hình về công cụ này.
- **Sinh các ca kiểm thử:** Đây là một trong những công cụ quan trọng nhất của các bộ kiểm thử tự động. Tùy thuộc vào các kỹ thuật kiểm thử được áp dụng, công cụ này sẽ sinh ra tập các ca kiểm thử (chứa gồm giá trị đầu ra mong muốn) cho chương trình/don vị chương trình cần kiểm thử. Các ca kiểm thử được sinh ra chỉ

	VIETTEL AI RACE	TD067
	TỔNG QUAN KIỂM THỬ TỰ ĐỘNG VÀ CÔNG CỤ HỖ TRỢ	Lần ban hành: 1

chứa giá trị đầu vào để thực hiện nó. Các giá trị này có thể được lựa chọn trong cơ sở dữ liệu hoặc được sinh một cách ngẫu nhiên.

- **Sinh giá trị đầu ra mong muốn:** Các ca kiểm thử được sinh ra bởi công cụ trên chỉ chứa các giá trị ứng với các biến đầu vào. Công cụ này cho phép sinh ra giá trị đầu ra mong muốn ứng với mỗi bộ dữ liệu đầu vào của mỗi ca kiểm thử. Giá trị đầu ra mong muốn này sẽ được sử dụng để so sánh với giá trị đầu ra thực tế khi thực hiện ca kiểm thử này nhằm phát hiện ra các lỗi/khiếm khuyết của sản phẩm.
- **So sánh kết quả kiểm thử:** Công cụ này so sánh giá trị đầu ra thực tế và giá trị đầu ra mong muốn của mỗi ca kiểm thử khi nó được thực hiện trên chương trình/đơn vị chương trình cần kiểm thử. Đối với mục đích kiểm thử phi chức năng, chúng ta không thể sử dụng cách làm này. Các giải pháp cho bài toán này sẽ được trình bày chi tiết trong chương 10.
- **Tạo báo cáo kiểm thử:** Một trong những ưu điểm của các bộ công cụ kiểm thử tự động là nó có cơ chế sinh báo cáo kiểm thử một cách chính xác và nhất quán. Dựa vào kết quả của công cụ so sánh kết quả kiểm thử, công cụ này sẽ tự động sinh ra báo cáo kết quả kiểm thử theo định dạng mong muốn của đơn vị phát triển.
- **Phân tích động:** Công cụ này cung cấp một cơ chế nhằm kiểm tra việc thực hiện của các câu lệnh của chương trình cần kiểm thử nhằm phát hiện ra các lỗi và phát hiện các câu lệnh/đoạn lệnh không được thực hiện bởi một tập các ca kiểm thử cho trước. Công cụ này cũng rất hiệu quả trong việc đánh giá tính hiệu quả của một bộ kiểm thử cho trước.
- **Bộ mô phỏng:** Có nhiều loại hình mô phỏng được cung cấp trong các bộ kiểm thử tự động. Mục đích của các công cụ này là mô phỏng quá trình thực hiện của chương trình cần kiểm thử. Ví dụ, các công cụ mô phỏng giao diện người dùng cho phép thực hiện tự động các tương tác giữa người dùng và sản phẩm. Selenium là một ví dụ về một công cụ mô phỏng giao diện người dùng cho các ứng dụng Web.

Trong thực tế, các bộ công cụ kiểm thử tự động có thể có thêm một số công cụ khác như cho phép đặc tả các tính chất của hệ thống cần kiểm thử, vân vân. Một số bộ công cụ chỉ hỗ trợ một số công cụ trong các công cụ đã liệt kê ở trên.

Các công cụ hỗ trợ kiểm thử tự động rất đa dạng và phục vụ nhiều mục đích khác nhau. Hình 9.2 thể hiện các loại công cụ kiểm thử tự động ứng với từng pha trong quy trình phát triển phần mềm.

	VIETTEL AI RACE	TD067
	TỔNG QUAN KIỂM THỦ TỰ ĐỘNG VÀ CÔNG CỤ HỖ TRỢ	Lần ban hành: 1

- **Các công cụ quản lý:** các công cụ này phục vụ pha lập trình và quản lý quá trình này như quản lý phiên bản, quản lý thay đổi, v.v.
- **Các công cụ phân tích tĩnh:** các công cụ này phục vụ pha lập trình và cho phép phân tích mã nguồn để tìm các lỗi hay gấp.
- **Các công cụ phân tích độ phủ:** các công cụ này hỗ trợ pha kiểm thử đơn vị và cho phép phân tích độ bao phủ của một bộ kiểm thử đối với mã nguồn.
- **Các công cụ gỡ lỗi:** các công cụ này cho phép định vị các lỗi được phát hiện bởi một ca kiểm thử.
- **Các công cụ phân tích động:** các công cụ này hỗ trợ cả pha kiểm thử đơn vị và kiểm thử tích hợp. Chúng cho phép sinh ra các ca kiểm thử từ mã nguồn và thực thi chúng nhằm phát hiện các lỗi lập trình.
- **Các công cụ mô phỏng, kiểm thử hiệu năng:** các công cụ này trợ giúp kiểm thử tự động hệ thống và kiểm thử chấp nhận. Chúng cho phép kiểm thử một số yêu cầu về hiệu năng của hệ thống như tính hiệu quả trong sử dụng tài nguyên, tính an toàn và bảo mật, khả năng chịu tải, v.v.
- **Các công cụ thực thi và đánh giá các ca kiểm thử:** các công cụ này cho phép thực hiện/thực thi các ca kiểm thử và giám sát việc thực hiện từng ca kiểm thử ứng với bộ giá trị đầu vào, giá trị đầu ra mong muốn và giá trị đầu ra thực tế nhằm tạo báo cáo kiểm thử.