1. Các khái niệm:
2. Khái niệm:

* Là quá trình kiểm tra, quan sát phần mềm dựa trên đặc tả yêu cầu từ khách hàng
* Tiến hành khảo sát nhằm các bên liên quan đến ứng dụng các thông tin ứng dụng dưới quá trình kiểm thử
* Quá trình thực thi nhằm tìm kiếm lỗi
* Tiến hành kiểm tra các hàm và sự đúng đắn của phần mềm qua quá trình chạy thử
* Quá trình phân tích nhằm kiểm tra sự khác biệt giữa phần mềm và yêu cầu của khách hàng

1. Mục đích:

* Tìm ra các lỗi hay khuyết điểm phần mềm nhằm đảm bảo hiệu quả hoạt động tối ưu của phần mềm trong nhiều ngành khác nhau

1. Vai trò ( tầm quan trọng):

* Phát hiện lỗi sớm sẽ giảm thiểu chi phí phát triển
* Chắc chắn phần mềm dễ sử dụng
* Chắc chắn phần mềm đáng tin cậy
* Tăng sự hài lòng của khách hàng
* Giảm chi phí sản phẩm

1. Sự khác nhau giữa bug và defect:

* Bug: lỗi phát sinh trong quá trình thực thi so người lập trình sinh ra
* Defect: lỗi không thống nhất giữa phần mềm và khách hàng ( phát hiện khi ship cho khách hàng)
* Nguyên nhân gây ra defect:
* Thiếu tài liệu nghiệp vụ
* Thiếu/ chậm thời gian phát triển
* Thiếu hiểu biết về lĩnh vực, nghiệp vụ
* Bắt buộc về ngôn ngữ lập trình

1. Kiểm thử hộp đen:
2. Khái niệm:

* Là loại kiểm thử mà người kiểm thử hoàn toàn không quan tâm đến cấu trúc, hành vi bên trong phần mềm. Người kiểm thử chỉ quan tâm đến việc tìm ra các hiện tượng mà phần mềm không xử lý theo đùng đặc tả và vì thế dữ liệu kiểm thử sẽ xuất phát từ đặc tả.
* Cố gắng tìm các lỗi: các chức năng thiếu hoặc không đúng, lỗi giao diện, lỗi cấu trúc dữ liệu, lỗi thi hành, lỗi khởi tạo hoặc kết thúc…

1. Kiểm thử lớp tương đương:

* Là một kỹ thuật kiểm thử hộp đen chia miền đầu vào thành các lớp dữ liệu , từ đó suy dẫn ra các ca kiểm thử. Phương pháp này cố gắng xác định một ca kiểm thử mà làm lộ ra một lớp lỗi, do đó là giảm số lượng trường hợp kiểm thử.
* Phương pháp: 2 bước:
* Xác định các lớp tương đương
* Tạo test case ứng với lớp tương đương
* Ứng dụng: giảm số lượng test case đáng kể, nên phù hợp với các hệ thống có nhiều giá trị đầu vào và nhận giá trị trong các miền
* Hạn chế: chỉ ở mức đơn vị, tích hợp,, tất cả các yêu cầu là đầu vào hay đầu ra có thể phân chia theo yêu cầu của hệ thống.

1. Kiểm thử giá trị biên: (Boundary Value Testing)

* Là kiểm thử dựa trên giá trị biên của vùng dữ liệu hợp lệ, mục đích là hệ thống không chấp nhận các dữ liệu nằm ngoài vùng hợp lệ mà chỉ chấp nhận các dữ liệu bên trong( bao gồm cả biên)
* Phương pháp: 3 bước:
* Xác định lớp tương đương
* Xác định biên tương ứng của các lớp tương đương
* Tạo test case cho các mỗi biên bằng cách chọn một điểm trên biên, 1 điểm dưới biên, 1 điểm nằm trên biên
* Ứng dụng: giảm đáng kể số lượng test case mà đáng ra phải thực hiện, nên phù hợp với các hệ thống có nhiều dữ liệu đầu vào nhận giá trị trong các miền
* Hạn chế: chỉ ở mức đơn vị, tích hợp, tất cả các yêu cầu đầu vào phân chia và các biên có thể xác định dựa vào yêu cầu hệ thống

1. Kiểm thử hộp trắng:
2. Khái niệm:

* Cho phép kiểm tra cấu trúc bên trong phần mềm với mục đích đảm bảo các câu lệnh và điều kiện sẽ được thực hiện ít nhất một lần.
* Yêu cầu: người kiểm thử phải có kĩ năng, kiến thức nhất định để có thể thông hiểu code. Mất rất nhiều thời gian nếu mức độ kiểm thử là tích hợp hay là hệ thống. Vì vậy chỉ sử dụng kiểm thử đơn vị

1. Kiểm thử luồng dữ liệu:

* Là kĩ thuật kiểm thử hộp trắng được sử dụng để phát hiện các lỗi giá trị dữ liệu trong chương trình
* Sử dụng đồ thị luồng dữ liệu dể tìm lỗi
* Phương pháp: 4 bước
* Vẽ đồ thị luồng dữ liệu
* Chọn tiêu chuẩn kiểm thử theo luồng
* Xác định đường đi
* Tạo các test case cho các đường đi

1. Công cụ Automation Testing:
2. Selenium:

* Khái niệm: là một công cụ kiểm thử phần mềm tự động, gồm có 4 phần:
* IDE: công cụ cho phép chúng ta record/ play back 1 test script. Là một add-on hỗ trợ trên firefox, record trên firefox mà play back đa trình duyệt
* Grid: hệ thống hỗ trợ người dùng thực thi test script trên nhiều trình duyệt một cách song song mà không cần chỉnh sửa test script
* RC, Web Driver: là thư viện cho phép chúng ta lập trình test script trên nhiều ngôn ngữ lập trình khác nhau: python, java, c#...
* Đặc điểm:
* Mã nguồn mở
* Cộng đồng hỗ trợ
* Hỗ trợ nhiều ngôn ngữ
* Chạy trên nhiều hệ điều hành khác nhau
* Chạy test script ở background
* Không hỗ trợ Win app

1. Soup UI:

* Khái niệm: là công cụ kiểm thử mã nguồn mở API.
* Đặc điểm:
* Giao diện đơn giản, thân thiện, dễ sử dụng
* Kiểm thử chức năng: Hỗ trợ tính năng kéo thả, gỡ lỗi
* Kiểm thử bảo mật: Ngăn chặn SQL Injection để bảo đảm dữ liệu, thực hiện fuzzing scan và boundary scan để tránh các hành vi bất thường
* Kiểm thử tải: Kiểm thử khả năng chịu tải của ứng dụng web sử dụng loadUI, cho phép báo cáo chi tiết để nắm hiệu suất
* Hỗ trợ các giao thức và công nghê: SOAP, REST, HTML, JMS,…

1. Sahi:

* Khái niệm: Là công cụ kiemr thử tự động các ứng dụng Web
* Đặc điểm:
  + Chạy lại trên bất kỳ trình duyệt máy tính và trên cả các trình duyệt di động.
  + Sahi đặc biệt thích hợp cho quá trình duyệt, thử nghiệm web.
  + Sahi hoạt động tốt trong môi trường Agile
  + Hỗ trợ kết nối data. Có thể kết nối với database, Excel hoặc CSV file.

1. Watir:

* Khái niệm: Là 1 thư viện miễn phí trong Ruby như kết nối và đọc dữ liệu từ cơ sở dữ liệu, đọc bảng tính, kết xuất XML, mã nguồn của người dùng cũng có thể được cấu trúc hóa để tái sử dụng.
* 3 loại: water classic, webdriver, spec
* Đặc điểm:
* Hỗ trợ viết các test case đơn giản, linh hoạt, dễ đọc, dễ bảo trì
* Kiểm thử nhiều ứng dụng khác nhau
* Hỗ trợ các chức năng của một ngôn ngữ lập trình (kết nối, đọc dữ liệu từ CSDL)

1. Maven:

* Khái niệm: là chương trình quản lý dự án được sử dụng nhiều bởi Java developer, nó thực hiện các công việc như khởi tạo project, biên dịch, đóng gói, chạy test… một cách tự động và nhanh chóng. Nó đặc biệt hữu dụng với các dự án tầm trung và lớn.
* Đặc điểm:
  + Giúp các nhà phát triển Java dễ dàng hơn trong việc quản lý các dự án Java.
  + Cung cấp các tính năng như Unit Test, Integration Test
  + Cung cấp các cách thiết lập dự án nhanh chóng chỉ trong vài giây
  + Tạo một chuẩn thống nhất cho tất cả các dự án
  + Chạy các dự án song song (Maven 3)

1. Graddle:

* Định nghĩa: Là một công cụ tự động build project dựa trên các khái niệm của Apache Ant và Apache Maven, đồng thời giới thiệu một “domain specific language” dựa trên Groovy thay cho hình thức khai báo cấu hình truyền thống (XML) của project. Được tích hợp sẵn tự động trong  Android Studio để giúp những nhà phát triển tổ chức, quản lý và từng bước xây dựng ứng dụng của họ.
* Đặc điểm:
  + - Rất linh hoạt
    - Mạnh trong việc xây dựng các đa dự án
    - Quản lí các phụ thuộc (dependency) rất tốt (dựa trên Apache Ivy)
    - Sử dụng DST,Groovy là build script
    - Là một mô hình phong phú để miêu tả dự án

1. JUnit:

* Định nghĩa: Framework đơn giản dùng cho việc tạo các unit testing tự động. Cung cấp cho chúng ta các gói lớp có sẵn cho phép chúng ta viết các phương thức test một cách dễ dàng. Chạy các test có thể lặp đi lặp lại
* Đặc điểm:
  + - Xác nhận việc kiểm tra kết quả được mong đợi
    - Các Test Suite cho phép chúng ta dễ dàng tổ chức và chạy các test
    - Hỗ trợ giao diện đồ họa và giao diện dòng lệnh

1. Jenkins

* Khái niệm: Ứng dụng dùng để theo dõi các dự án phần mềm rất hiệu quả, nó giúp cho người dùng có thể liên tục cập nhật về tình hình của dự án thông qua procmail. Dùng để kiểm tra các lỗi bug và fix kịp thời khi mà người dùng chỉnh sửa và upload source code lên server dẫn đến việc trùng lặp hay thiếu các thư viện của project.
* Đặc điểm:
  + Dễ dàng cài đặt và sử dụng.
  + Đa nền tảng.
  + Hỗ trợ cho nhiều công nghệ phát triển phần mềm.
  + Quá trình build và test hoàn toàn tự động.
  + Dễ dàng liên kết với các công cụ khác của hệ thống tích hợp liên tục thông qua các plug in.
  + Miễn phí.

1. Sonar:

* Định nghĩa: là một nền tảng mã nguồn mỡ dùng để quản lý chất lượng mã nguồn. Nó liên tục phân tích và đo lường chất lượng kỹ thuật, từ danh mục dự án cho đến các phương pháp
* Đặc điểm:
  + là một ứng dụng dựa trên nền web, cung cấp các công cụ giúp phân tích mã nguồn, tìm kiếm báo cáo lỗi.
  + Cung cấp hơn 50 plugins để cộng đồng phát triển có thể mở rộng chức năng.

1. So sánh các tool:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tool/Tiêu chí | Selenium | SOAP UI | Sahi | Watir |
| Nền tảng | -Chạy trên nhiều trình duyệt ( trừ IDE có record trên firefox) | - có trình thuật sĩ (Wizard Setup) riêng để cài chương trình, sau khi cài sử dụng như các chương trình windows bình thường (sử dụng tệp .exe) | - Chạy trên nhiều trình duyệt khác nhau (IE,FF,Chrome,Opera,Safari) - Sử dụng các thuật toán khác nhau đê xác định các phần tử mà con người dễ dàng nhận thấy đc | Chạy trên các trình duyệt Firefox, Chrome, IE, Opera |
| Ngôn ngữ | Java, Ruby, Perl, Python, C# (and may be more). | - Groovy, JavaScript | Sahi Script, Java, Ruby Sahi Script, | ASP.Net, JSP, PHP, Rails |
| Tính dễ sử dụng | - Thời gian cài đặt dưới 5 phút vì sử dụng plugin của firefox - Cần kiến thức ngôn ngữ lập trình | - Giao diện trực quan, dễ nắm bắt - cần có kiến thức chuyên sâu về API để có thể sử dụng thành thạo | - Cần cài đặt, thời gian từ 10-30 phút, phụ thuộc vào lúc cài đặt java - Hỗ trợ APIs cho các task phức tạp | Hỗ trợ test case đơn giản, linh hoạt, dễ đọc |
| Tính ổn định của script và dễ dàng bảo trì | - Chạy test script trên background - Khó khăn cho tester trong việc hiểu và sửa lỗi | ổn định | - Không phụ thuộc XPath - Sử dụng \_near, and \_in | Dễ bảo trì |
| Sự phụ thuộc vào các tool khác | - Cần JUnit để chạy | - là tool trên nền java (ver 1,6+) nếu lúc cài không có cũng không sao, khi cài nó sẽ tự cài JRE v - nếu có jre rồi thì phải set JAVA\_HOME đàng hoàng. | - Không cần tool khác. Có thể chạy trên sahi controller, command line, ant | ? |
| Báo cáo | - Cần tool ngoài để đọc được báo cáo | - chỉ có trong bản Pro, - được generate từ các các level Project, TestSuite, TestCase,LoadTest - hỗ trợ xuất ra 3 dạng: + định dạng in được: PDF, HTML, RTF, Excel,... các định dạng này đều là customizable + data export: định dạng XML, csv, phù hợp để import vào các tool khác, hay để integration + HTML report : báo cáo tổng quan các kết quả dưới dạng HTML (ko hỗ trợ ở level LoadTest) | - Có sẵn HTML report được build dưới dạng script | Report file excel ( phương thức ruby), html |
| Hỗ trợ có sẵn | - Có hỗ trợ miễn phí trên forum - Hỗ trợ trả phí | - Tutorial nhiều vô số | - Có hỗ trợ miễn phí trên forum - Hỗ trợ trả phí | - free  - hỗ trợ trả phí |
| Hệ điều hành | - Hỗ trợ chạy trên nhiều hệ điều hành khác nhau | - tất cả các hệ điều hành hỗ trợ JRE | - Hỗ trợ chạy trên nhiều hệ điều hành khác nhau | ? |

|  |  |  |
| --- | --- | --- |
| Tiêu chí | Maven | Gradle |
| Tính dễ học | - là công cụ dựa trên xml nên dễ học | - dựa trên hệ thống DSL(công nghệ thuê bao số) nên khó học |
| Tốc độ | Tương đương | Tương đương |
| Tính phức tạp | Xây dựng theo XML, cấu trúc được xác định trước, chuẩn hơn nhưng ít linh hoạt | Chặc chẽ trong kết nối, linh hoạt, đơn giản và ngắn hơn |
| Số plugin và cách tùy chỉnh | Có hàng tram plugins và đơn giản trong việc tạo 1 plugin | Dựa trên plugin chung, tạo ra thì dễ mà thống nhất, chính xác thì khó |
| Tài liệu | Lâu đời, tài liệu tốt, nhiều nhân lưc, có sự giúp đỡ của các diên đàn | Mới, mã nguồn mở nhưng dưới dự kiểm soát của gradleware |
| Công cụ tích hợp | Hỗ trợ hầu hết các công cụ và nền tảng ( IDE, App Server, Cl) | Không hỗ trợ App server và Cl |