* Kiểm thử phần mềm là hoạt động khảo sát thực tiễn sản phẩm hay dịch vụ phần mềm trong đúng môi trường chúng dự định sẽ được triển khai nhằm cung cấp cho khách hàng những thông tin về chất lượng của sản phẩm hay dịch vụ phần mềm ấy
* Mục tiêu kiểm thử là tìm ra các lỗi hay khiếm khuyết phần mềm nhằm đảm bảo hiệu quả hoạt động tối ưu của phần mềm trong nhiều ngành khác nhau.
* Vai trò của kiểm thử phần mềm: Kiểm thử phần mềm đóng vai trò quan trọng trong việc đánh giá và thu được chất lượng cao của sản phẩm phần mềm trong quá trình phát triển. Kiểm tra sự khác biệt giữa phần mềm và yêu cầu của khách hàng. Thông qua chu trình “*kiểm thử - tìm* *lỗi - sửa lỗi*”, ta hy vọng chất lượng của sản phẩm phần mềm sẽ được cải tiến.
* Bug # defect:
  + Bug: lỗi phát sinh trong quá trình thực thi do người lập trình phát sinh ra.
  + Defect: lỗi không thống nhất giữa phần mềm và yêu cầu của người dùng, phát hiện khi đã chuyển tay đến khách hàng.
  + Nguyên nhân gây ra defect:
    - Thiếu tài liệu của yêu cầu nghiệp vụ
    - Thiếu hiểu biết về lĩnh vực nghiệp vụ
    - Do bắt buộc về ngôn ngữ lập trình
    - Thiếu hoặc chậm thời gian phát triển
* Tầm quan trọng của testing
  + Phát hiện lỗi sớm sẽ giảm thiểu chi phí
  + Phần mềm chắn chắn đáng tin cậy và dễ sử dụng
  + Tăng sự hài lòng của khách hàng
  + Giảm chi phí sản phẩm
* Kiểm tra hộp đen là phương pháp kiểm thử mà người kiểm thử hoàn toàn không quan tâm đến cấu trúc bên trong phần mềm, mà chỉ quan tâm đến dữ liệu vào ra xử lý theo đúng đặc tả của nó.
* Phân tích giá trị biên (BVA) là một kỹ thuật kiểm thử hộp đen mà tập trung chủ yếu vào vùng biên của miền xác định (những giá trị đặc biệt của biến đầu vào) việc xây dựng những test case
* Phân lớp tương đương là một kỹ thuật kiểm thử hộp đen Phân chia các testcase thành các lớp (họ, tập) có cùng tính chất với mục đích giảm số lượng testcase nhưng không giảm chất lượng của kiểm thử, từ đó một phần tử bất kỳ trong một lớp đều có thể làm đại diện cho một lớp với một trường hợp cụ thể.
* Kiểm thử hộp trắng hay còn gọi là kiểm thử hướng logic, cho phép kiểm tra cấu trúc bên trong của phần mềm với mục đích đảm bảo tất cả các câu lệnh và đk sẽ được thực hiện ít nhất 1 lần. Trong kỹ thuật này người kiểm thử lấy dữ liệu xuất phát từ việc kiểm tra logic của chương trình.
* Kiểm thử luồng dữ liệu là một kỹ thuật kiểm thử hộp trắng có thể được sử dụng để phát hiện sử dụng không đúng cách của các giá trị dữ liệu trong chương trình.
* Kiểm chứng luồng dữ liệu sử dụng đồ thị luồng điều khiển để tìm những điều bất hợp lý có thể xảy ra với dữ liệu.

|  |  |  |  |
| --- | --- | --- | --- |
| Selenium | SOAP UI | Sahi | Watir |
| Đều là những công cụ kiểm thử ứng dụng web tự động, là những công cụ mã nguồn mở, có thể triển khai trên đa nền tảng như: Windows, Ubuntu,… | | | |
| Selenium IDE cho phép Record/ Playback một test script.  Cho phép lưu test case đã Record dưới nhiều ngôn ngữ lập trình khác nhau.  Cho phép thực thi kiểm thử song song trên nhiều máy khác nhau vs nhiều trình duyệt khác nhau  Không có khả năng test GUI | Tính năng kéo thả là tăng tốc độ phát triển script  Cho phép tùy chỉnh báo cáo chi tiết để nắm bắt các thông số hiệu suất  Ngăn chặn SQL Injection để đảm bảo CSDL  Kiểm thử khả năng chịu tải của 1 ứng dụng loadUI  Cho phép bắt chước, tạo ra các bài kiểm tra mạnh mẽ đối với các dịch vụ Web trước khi chúng thực hiện. |  | 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. |

**\*Gradle**

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.

**\*Mayven**

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

**\*Junit**

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

**\*Jenkins**

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.

**\*Sonar**

SonarQube quản lý 7 khía cạnh kỹ thuật khi thực hiện phân tích mã nguồn một dự án, nếu vi phạm sẽ phá hỏng code của bạn và được các nhà phát triển gọi **7 tội lỗi chết người (**seven deadly sins):

* + Lỗi và lỗi tìm năng
  + Vi phạm các tiêu chuẩn code
  + Trùng lắp
  + Thiếu các Unit Test
  + Những thành phần liên kết phức tạp
  + Thiết kế lộn xộn
  + Không đủ hoặc quá nhiều comment