# TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM

## Khái niệm và mục đích của kiểm thử

### Khái niệm

Kiểm thử phần mềm là một quá trình thực thi một chương trình với mục đích tìm ra lỗi. Kiểm thử phần mềm bảo đảm sản phẩm phần mềm đáp ứng chính xác, đầy đủ và đúng theo yêu cầu của khách hàng, yêu cầu sản phẩm đã đặt ra.

Kiểm thử phần mềm cũng cung cấp mục tiêu, quan điểm độc lập về phần mềm, điều này cho phép việc đánh giá và hiểu rõ các rủi ro khi thực thi phần mềm. Kiểm thử phần mềm tạo điều kiện để tận dụng tối đa tư duy đánh giá và sáng tạo để bạn có thể phát hiện ra những điểm mà người khác chưa nhìn thấy. Người thực hiện kiểm thử được gọi là kiểm thử viên (Tester).

Kiểm thử phần mềm bao gồm:

* Kiểm thử thủ công (Manual Testing): Là quá trình kiểm thử phần mềm mà tester kiểm thử bằng tay bao gồm viết kịch bản kiểm thử, nhập dữ liệu và thực hiện kiểm thử. Hiện nay, phần lớn các tổ chức và các công ty phần mềm đều kiểm thử thủ công là chủ yếu.
* Kiểm thử tự động (Automation Testing): Là quá trình sử dụng các công cụ và phần mềm để thực hiện các thử nghiệm kiểm tra một cách tự động, thay vì phải thực hiện thủ công. Các kịch bản kiểm thử được tự động hóa và thực thi bởi các công cụ phần mềm, giúp cho việc kiểm tra tính đúng đắn của ứng dụng phần mềm trở nên hiệu quả hơn và tiết kiệm thời gian.

### Mục đích của kiểm thử phần mềm

Kiểm thử phần mềm là một trong những yếu tố quyết định chất lượng của phần mềm. Thông thường thì phần mềm không hoạt động như mong muốn dẫn đến việc lãng phí thời gian, tiền bạc, uy tín của doanh nghiệp, thậm chí có thể gây nên tổn thất, thiệt hại vô cùng lớn cho cá doanh nghiệp.

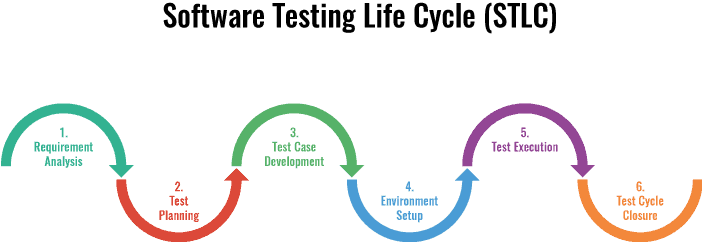
Mục tiêu chính của kiểm thử phần mềm:

* Ngăn ngừa lỗi bằng cách đánh giá các chức năng như yêu cầu và thiết kế.
* Xác minh xem tất cả các yêu cầu đã đáp ứng hay chưa.
* Kiểm tra xem đối tượng thử nghiệm đã hoàn thành chưa và xác nhận.
* Xây dựng niềm tin về mức độ chất lượng.
* Tìm ra lỗi, khiếm khuyết để giảm rủi ro của chất lượng phần mềm không đầy đủ.
* Cung cấp đủ thông tin cho các bên liên quan để họ có thể đưa ra quyết định sáng suốt, đặc biệt về chất lượng của đối tượng thử nghiệm.
* Tuân thủ các yêu cầu hoặc tiêu chuẩn về hợp đồng, pháp lý.
* Tạo các test case chất lượng cao, thực hiện kiểm thử hiệu quả và tạo ra các báo cáo vấn đề đúng và hữu dụng.

## Quy định kiểm thử phần mềm

Tùy vào từng tổ chức, hệ thống, ngữ cảnh, mức độ rủi ro của phần mềm mà quy trình kiểm thử phần mềm có thể gồm nhiều bước khác nhau. Nhưng nhìn chung mọi quy trình kiểm thử đều có những bước cơ bản sau:

* Phân tích yêu cầu (Requirement Analysis).
* Lập kế hoạch kiểm thử (Test Planning)
* Phát triển kịch bản kiểm thử (Test Case Development)
* Thiết lập môi trường kiểm thử (Environment Setup)
* Thực hiện kiểm thử (Test Execution )
* Kết thúc chu kỳ kiểm thử **(**Test Cycle Closure)



*Hình 1. 1: Quy trình kiểm thử phần mềm*

### Phân tích yêu cầu

Phân tích yêu cầu là giai đoạn đầu tiên trong quy trình kiểm thử phần mềm, gồm các hoạt động sau:

* Tìm hiểu, nghiên cứu và phân tích cụ thể các yêu cầu trong tài liệu đặc tả của dự án hoặc tài liệu khách hàng.
* Đưa ra câu hỏi với các bên liên quan như chuyên viên phân tích nghiệp vụ (BA), người quản lý dự án (PM), trưởng nhóm và khách hàng để hiểu chính xác hơn về yêu cầu của sản phẩm.

### Lập kế hoạch kiểm thử

Lập kế hoạch kiểm thử theo các bước quan trọng sau:

* Xác định phạm vi, rủi ro và mục đích của hoạt động kiểm thử.
* Xác định cách tiếp cận kiểm thử.
* Xác định quy định kiểm thử hoặc chiến lược kiểm thử.
* Xác định yêu cầu về nguồn nhân lực như con người, môi trường kiểm thử, thiết bị,...
* Lên lịch trình cho việc phân tích kiểm thử và thiêt kế các trường hợp. kiểm thử, thực thi kiểm thử và đánh giá kết quả kiểm thử.
* Xác định các tiêu chí kết thúc việc kiểm thử.

### Thiết kế kịch bản kiểm thử

Thiết kế kịch bản kiểm thử là bước hết sức quan trọng, gồm các hoạt động sau:

* Kiểm tra lại tất cả các tài liệu để xác định công việc cần làm, các công việc có khác gì so với dự án trước khách hàng đưa cho.
* Viết kịch bản kiểm thử (test case hoặc checklist).
* Chuẩn bị dữ liệu kiểm thử.
* Thành viên trong đội kiểm thử đánh giá lại test case/checklist.

### Thiết lập môi trường kiểm thử

Việc cài đặt môi trường kiểm thử là giai đoạn cũng rất quan trọng trong vòng đời phát triển phần mềm, gồm các hoạt động sau:

* Quyết định môi trường kiểm thử.
* Chuẩn bị một số test case để kiểm tra xem môi trường cài đặt đã sẵn sàng cho việc kiểm thử hay chưa.

### Thực hiện kiểm thử

Thực hiện kiểm thử gồm các hoạt động sau:

* Thực hiện các test case như thiết kế và mức độ ưu tiên đã đưa ra trên môi trường đã được cài đặt.
* So sánh với kết quả mong đợi sau báo cáo các lỗi xảy ra lên công cụ quản lý lỗi.
* Thực hiện kiểm thử lại (re-test) để xác minh các lỗi đã được sửa.
* Đo và phân tích tiến độ.
* Báo cáo thường xuyên cho người quản lý dự án (PM) và khách hàng về tình hình thực hiện dự án.

### Đóng chu trình kiểm thử

Đây là giai đoạn cuối cùng trong quy trình kiểm thử phần mềm, gồm các hoạt động sau:

* + Tổng kết, báo cáo kết quả về việc thực thi test case.
  + Đánh giá các tiêu chí hoàn thành như phạm vi kiểm tra, chất lượng, chi phí, thời gian, mục tiêu kinh doanh quan trọng.
  + Thảo luận tất cả những điểm tốt, điểm chưa tốt và rút ra bài học kinh nghiệm.

## Phương pháp kiểm thử phần mềm

### Kiểm thử hộp đen (Black Box Testing)

Kỹ thuật kiểm thử hộp đen hay còn gọi là kiểm thử vào/ra. Trong kỹ thuật này, người kiểm thử xem phần mềm như là một hộp đen. Người kiểm thử hoàn toàn không quan tâm đến cấu trúc bên trong của chương trình mà chỉ quan tâm tới dữ liệu đầu vào và đầu ra sau khi được xử lý vì thế dữ liệu kiểm thử sẽ xuất phát từ đặc tả.

Kiểm thử hộp đen sẽ cố gắng tìm được các lỗi về: Giao diện, các chức năng thiếu hoặc không đúng, các lỗi cấu trúc dữ liệu trong việc truy cập cơ sở dữ liệu bên ngoài, lỗi thi hành, các lỗi khởi tạo hoặc kết thúc.

Kiểm thử hộp đen có ưu và nhược điểm như sau:

* + Ưu điểm: Kiểm thử hộp đen không yêu cầu kiến thức sâu về cấu trúc bên trong của phần mềm và có thể thực hiện mà không cần biết mã nguồn hoặc cấu trúc nội bộ của ứng dụng.
  + Nhược điểm: Hạn chế sâu về kiến thức về cấu trúc và logic nội bộ của ứng dụng. Khó khăn trong việc tìm ra và khắc phục lỗi do không có thông tin chi tiết về cách hoạt động bên trong của phần mềm.

### Kiểm thử hộp trắng (White Box Testing)

Kiểm thử hộp trắng hay còn gọi là kiểm thử logic là một phương pháp kiểm thử phần mềm trong đó tester biết về cấu trúc nội bộ thiết kế. Người kiểm tra chọn đầu vào để thực hiện các đường dẫn thông qua mã và xác định đầu ra thích hợp. Kiến thức lập trình và kiến thức thực hiện là rất cần thiết trong kiểm thử hộp trắng.

Kiểm thử hộp trắng có ưu và nhược điểm sau:

* + Ưu điểm: Hiểu biết sâu về cấu trúc nội bộ của phần mềm, giúp phát hiện và khắc phục các lỗi ở mức độ chi tiết. Có thể tối ưu hóa hiệu suất và tính ổn định của ứng dụng từ cấu trúc bên trong.
  + Nhược điểm: Đòi hỏi kiến thức kỹ thuật sâu về mã nguồn và cấu trúc phần mềm. Yêu cầu nhiều thời gian và nguồn lực để thực hiện và duy trì quá trình kiểm thử.

### Kiểm thử hộp xám (Grey Box Testing)

Là phương pháp kiểm thử khá mới mẻ, mới hình thành và đòi hỏi trình độ cao. Kiểm thử hộp xám là kiểu trung gian giữa hộp đen và hộp trắng. Đòi hỏi người kiểm thử phải vận dụng kiến thức về thuật toán, cấu trúc bên trong của chương trình như là hộp trắng nhưng để thiết kế ca kiểm thử theo hướng người sử dụng, hoặc có ca kiểm thử như của hộp đen.

Kiểm thử hộp xám có ưu và nhược điểm sau:

* + Ưu điểm: Tiếp cận từ góc độ người dùng, giúp đảm bảo tính toàn diện của kiểm thử. Dựa trên các đặc tả chức năng, mô tả người dùng và sơ đồ kiến trúc, giúp xác nhận yêu cầu sớm. Kết hợp lợi ích của cả kiểm thử hộp đen và hộp trắng.
  + Nhược điểm: Có thể tốn nhiều thời gian để kiểm tra từng luồng, đôi khi không thực tế. Phạm vi kiểm tra thường thấp hơn so với kiểm thử hộp trắng và hộp đen riêng biệt.

## Các mức kiểm thử

Mức kiểm thử được hiểu là giai đoạn kiểm thử gắn liền với hoạt động của quy trình kiểm thử và liên quan chặt chẽ tới quy trình sản xuất của một phần mềm.

Số lượng mức kiểm thử thường cố định ở 4 mức:

* Kiểm thử đơn vị (Unit Test)
* Kiểm thử tích hợp (Integration Test)
* Kiểm thử hệ thống (System Test)
* Kiểm thử chấp nhận (Acceptance Test)

### Kiểm thử đơn vị (Unit test)

Unit Testing là giai đoạn đầu tiên trong kiểm thử phần mềm. Unit Testing được thực hiện nhằm kiểm tra và xác định các module riêng lẻ thuộc mã nguồn có hoạt động đúng hay không.

Mục đích: Đảm bảo chức năng của đơn vị hoạt động đúng theo mong muốn. Người thực hiện: Do việc kiểm thử đơn vị đòi hỏi phải kiểm tra từng nhánh lệnh, nên đòi hỏi người kiểm thử có kiến thức về lập trình cũng như về thiết kế của hệ thống nên người thực hiện thường là lập trình viên.

### Kiểm thử tích hợp (Integration Test)

Kiểm thử tích hợp cấp độ kiểm thử phần mềm tích hợp của các đơn vị riêng lẻ được kết hợp và thử nghiệm thành một nhóm thông qua việc tập trung vào kiểm tra truyền dữ liệu giữa các module.

Mục đích: Phát hiện lỗi giao tiếp xảy ra giữa các Unit cũng như lỗi của bản thân từng Unit. Ngoài ra, mục đích là tích hợp các Unit đơn lẻ thành các hệ thống nhỏ và cuối cùng là nguyên hệ thống hoàn chỉnh chuẩn bị cho kiểm tra ở mức hệ thống.

Người thực hiện: Nhóm nhân viên kiểm thử.

### Kiểm thử hệ thống (System test)

Kiểm thử hệ thống là giai đoạn thứ 3 của kiểm thử phần mềm cho phép phần mềm hoàn chỉnh và tích hợp được kiểm tra. System Testing tập trung nhiều hơn vào các chức năng của toàn bộ hệ thống. Kiểm thử hệ thống bao gồm kiểm thử chức thăng và kiểm thử phi chức thăng.

Mục đích: Kiểm tra xem hệ thống được làm ra có thỏa mãn yêu cầu hay không về nhiều khía cạnh: Hoạt động, độ tin cậy, hiệu năng hệ thống.

Người thực hiện: Nhóm nhân viên kiểm thử.

### Kiểm thử chấp nhận (Acceptance test)

Kiểm thử chấp nhận được thực hiện bởi khách hàng hoặc ủy quyền cho nhóm thứ ba nhằm kiểm tra hệ thống vừa xây dựng đã phù hợp với yêu cầu của khách hàng trước đó hay chưa.

Mục đích: Mục tiêu chính đằng sau kiểm thử chấp nhận là để kiểm tra xem sản phẩm phần mềm được phát triển có vượt qua các tiêu chuẩn chấp nhận được xác định trên cơ sở yêu cầu của người dùng và doanh nghiệp hay không, để tuyên bố rằng người dùng có thể chấp nhận hoặc không chấp nhận sử dụng sản phẩm đó.

Người thực hiện: Khách hàng.

## Kỹ thuật viết kịch bản kiểm thử

### Kỹ thuật phân vùng tương đương

Kỹ thuật phân vùng tương đương (Equivalence Partitioning – EP) là loại thiết kế test dạng hộp đen (Black box) mà trong đó, các test case sẽ được thiết kế là đại diện cho những phân vùng tương đương.

Phân vùng tương đương là một phân đoạn của miền giá trị đầu và hoặc đầu ra mà tại đó, các thành phần hoặc hệ thống sẽ có hoạt động giống nhau.

Để thực hiện phân vùng cho các lớp tương đương, chúng ta có hai bước chính sau đây:

* Bước 1: Xác định các phân lớp tương đương của đầu vào và đầu ra. Chọn từng điều kiện của đầu vào và đầu ra được mô tả trong spec, chia làm hai hai phân lớp:
* Phân lớp thỏa màn điều kiện – Phân lớp hợp lệ.
* Phân lớp không thỏa mãn điều kiện – Phân lớp không hợp lệ.
* Bước 2: Dựa trên các phân lớp đã chia, lựa chọn testcase tương ứng.

### Kỹ thuật phân tích giá trị biên

Phân tích giá trị biên (Boundary Value Analysis - BVA) là kỹ thuật thiết kế test case nhằm kiểm thử những giá trị biên giữa các phân vùng với nhau.

Có hai cách tiếp cận BVA:

* Kiểm thử hai giá trị: giá trị ranh giới (trên đường biên) và giá trị nằm vượt quá chút đường biên (theo gia số nhỏ nhất có thể) được sử dụng.
* Kiểm thử ba giá trị: Sử dụng giá trị trước biên, giá trị biên và giá trị vượt qua biên.

Nếu các giá trị biên thuộc phân lớp hợp lệ thì chúng là những giá trị hợp lệ. Ngược lại nếu các giá trị biên thuộc phân lớp không hợp lệ thì chúng là những giá trị không hợp lệ.

Các test case có thể được thiết kế để bao phủ cả những giá trị hợp lệ và giá trị không hợp lệ. Khi thiết kế test case, mỗi giá trị biên được chọn sẽ có một trường hợp kiểm thử.

### Kỹ thuật đoán lỗi

Một kỹ thuật thiết kế ca kiểm thử khác là đoán lỗi. Kiểm thử viên phỏng đoán lỗi dựa trên trực giác và kinh nghiệm của mình, từ đó liệt kê các trường hợp có thể xảy ra lỗi và sinh ca kiểm thử.

Khó có thể đưa ra một quy trình cho kỹ thuật kiểm thử đoán lỗi vì nó có tính trực giác cao và không thể dự đoán trước. Trong một số trường hợp, kiểm thử viên có thể kết hợp với lập trình viên để tìm ra những trường hợp có thể bị bỏ sót trong quá trình viết đặc tả yêu cầu phần mềm và lập trình.

### Kỹ thuật chuyển đổi trạng thái

Kỹ thuật này dựa trên việc quan sát, theo dõi quá trình chuyển từ trạng thái này sang trạng thái khác khi có một hành động xảy ra trong chương trình phần mềm để phát hiện các lỗi có thể xảy ra mà các kỹ thuật trên có thể bỏ sót.

Ví dụ điển hình cho kỹ thuật chuyển trạng thái là việc kiểm thử chức năng giỏ hàng trong các trang Web thương mại điện tử. Lỗi có thể xuất hiện mỗi khi thêm sản phẩm vào giỏ hàng, xóa sản phẩm khỏi giỏ hàng hay khi thanh toán các sản phẩm trong giỏ hàng đó.

Công việc của kiểm thử viên là xem xét các điều kiện trạng thái, theo dõi quá trình chuyển đổi giữa các trạng thái, điều kiện nhập đầu vào và các sự kiện kích hoạt thay đổi trạng thái.

## Kiểm thử trên nền tảng web

### Khái quát về kiểm thử trên nền tảng web

Khi mạng Internet ngày càng phát triển, việc thiết kế website và các ứng dụng chạy trên nền web trở nên cần thiết để tận dụng cơ hội kinh doanh và tiếp cận khách hàng. Các ứng dụng web đóng vai trò quan trọng trong việc kết nối và trao đổi thông tin của nhiều doanh nghiệp.

Để đạt được thành công, các ứng dụng web cần có chất lượng cao và hiệu suất tốt, bao gồm cả giao diện và trải nghiệm người dùng. Ngoài ra, ứng dụng web cũng có các đặc thù riêng biệt so với ứng dụng di động hoặc desktop. Chúng có thể chạy trên nhiều nền tảng khác nhau và đòi hỏi các yêu cầu cấu hình và thao tác đặc biệt. Điều này đặt ra thách thức lớn đối với các nhà phát triển phần mềm trong việc đảm bảo chất lượng của ứng dụng web khi chạy trên nhiều nền tảng.

Vì vậy, việc phát triển một chiến lược kiểm thử hiệu quả là cần thiết để giảm thiểu rủi ro và nâng cao chất lượng của ứng dụng web.

### Các hạng mục kiểm thử Website

Kiểm thử chức năng: Kiểm tra tính đúng đắn và hiệu quả của các chức năng trong ứng dụng web, bao gồm các liên kết, form nhập liệu, tìm kiếm và các tính năng khác. Mục tiêu là phát hiện và báo cáo về bất kỳ lỗi nào xuất hiện khi sử dụng các chức năng này.

Kiểm thử giao diện: So sánh thiết kế UI với thực tế trên trình duyệt, kiểm tra mỗi thành phần trên trang web để đảm bảo đúng với bản thiết kế. Kiểm tra các liên kết và menu, đảm bảo chúng hoạt động đúng và không trỏ về chính nó.

Kiểm thử cookie và session: Kiểm tra tính an toàn của việc đăng nhập bằng cách thử các tập tin cookie khác nhau và kiểm tra khả năng bảo mật của ứng dụng khi xóa các tập tin cookie.

Kiểm thử nội dung đa ngôn ngữ: Đảm bảo dữ liệu được dịch sang các ngôn ngữ khác nhau mà không mất đi ý nghĩa hoặc gây ra lỗi.

Kiểm thử cơ sở dữ liệu: Kiểm tra kết nối và truy vấn cơ sở dữ liệu để đảm bảo dữ liệu chính xác và các chức năng hoạt động đúng.

### Một số công cụ hỗ trợ kiểm thử Website

* Công cụ kiểm thử hiệu năng:
* WebLoad: Cho phép kiểm tra khả năng chịu tải và độ chịu lỗi của ứng dụng web bằng nhiều công nghệ khác nhau.
* Apache JMeter: Công cụ mã nguồn mở để kiểm tra hiệu năng và chức năng của ứng dụng web.
* NeoLoad: Đo và phân tích hiệu suất của ứng dụng web bằng cách tăng lưu lượng truy cập.
* Công cụ kiểm thử bảo mật:
* Burp Suite: Công cụ kiểm tra lỗ hổng bảo mật cho ứng dụng web với nhiều tính năng tích hợp như Spider và Intruder.
* OWASP Zed Attack Proxy: Đánh giá an ninh mạng và bảo mật của các ứng dụng web.
* Nikto: Máy quét lỗ hổng máy chủ web mã nguồn mở để phát hiện cài đặt và cấu hình lỗi.
* Công cụ kiểm thử chức năng:
* BrowserStack: Kiểm thử chức năng của ứng dụng web trên nhiều trình duyệt khác nhau và kiểm tra khả năng hiển thị responsive.
* Ranorex: Công cụ tự động kiểm thử cho ứng dụng web, desktop và di động.
* Selenium: Công cụ mạnh mẽ cho kiểm thử tự động ứng dụng web, được phát triển và hỗ trợ bởi Selenium team từ Google.

## Kết luận

Trong chương 1 của đồ án em đã trình bày được các vấn đề cơ bản của kiểm thử phần mềm và kiểm thử trên nền tảng web.

* Khái niệm và mục đích của kiểm thử phần mềm.
* Quy trình kiểm thử phần mềm.
* Phương pháp kiểm thử phần mềm.
* Các mức kiểm thử.
* Kỹ thuật viết kịch bản kiểm thử.
* Kiểm thử trên nền tảng web.

Trong chương 1 đề cập đến kiểm thử tự động, giới thiệu tổng quan về công cụ Selenium, đi sâu vào giới thiệu kiến trúc và hướng dẫn cài đặt Selenium WebDriver.