# 1. Kỹ thuật kiểm thử tĩnh - Static Techniques

## 1.1. Định nghĩa:

* Kiểm thử tĩnh – Static testing : Kiểm thử một thành phần hoặc hệ thống theo đặc tả mà

không cần thực hiện phần mềm.

Ví dụ như đánh giá (Review) hoặc phân tích tĩnh (static analysis).

* Kỹ thuật kiểm thử tĩnh dựa trên việc kiểm tra thủ công (reviews) và phân tích tĩnh tự động

(static analysis) của mã (code) hoặc tài liệu dự án mà không thực thi mã chương trình.

Phương pháp kiểm thử tĩnh nhằm tìm kiếm những thiếu sót hơn là tìm kiếm những hoạt động không mong đợi, làm tăng chất lượng của phần mềm thông qua việc tìm lỗi ở giai đoạn đầu của quá trình phát triển. Phương pháp này thường tốn ít chi phí hơn so với kiểm thử động nhất là khi kiểm thử hồi quy.

## 1.2. Reviews

### 1.2.1.

* Đánh giá (reviews) là một cách kiểm thử sản phẩm công việc phần mềm ( bao gồm cả code) được thực hiện trước kiểm thử động. Các lỗi được phát hiện trong quá trình review sớm trong chu trình vòng đời ( ví dụ các lỗi được tìm thấy trong đặc tả yêu cầu) rẻ hơn nhiều so với các lỗi được phát hiện bằng cách chạy kiểm thử thi hành các mã.
* Một đánh giá (review) được thực hiện như thế nào?
  + Đánh giá thủ công (Review manually)
  + Dùng công cụ hỗ trợ (Automated Analysis by Tool).
* Đối tượng đánh giá bao gồm :
  + Các chi tiết thiết kế (design specifications)
  + Mã (Code)
  + Kế hoạch kiểm thử (test plans)
  + Kỹ thuật kiểm thử (test specifications)
  + Các trường hợp kiểm thử (test cases)
  + Các kịch bản kiểm thử (test scripts)
  + Hướng dẫn sử dụng (user guides)
  + Các trang web (web pages).
* Lợi ích của đánh giá (review)
  + Sớm phát hiện lỗi và điều chỉnh
  + Nâng cao chất lượng và năng suất của phát triển phần mềm.
  + Giảm khoảng thời gian phát triển
  + Giảm chi phí và thời gian kiểm thử
* Loại lỗi trong đánh giá (reviews) bao gồm:
  + Độ lệch tiêu chuẩn (deviations from standards)
  + Lỗi trong yêu cầu ( requirement defects)
  + Lỗi thiết kế ( design defects)
  + Không đủ khả năng bảo trì code (insufficient maintainability)
  + Chi tiết kỹ thuật giao diện không chính xác (incorrect interface specifications).
* Đánh giá có thể tìm các thiếu sót mà không có khả năng tìm thấy trong kiểm thử động. Ví dụ như thiếu sót trong yêu cầu kỹ thuật.

### 1.2.2. Quy trình đánh giá - Review Process

Các định nghĩa:

* Tiêu chí đầu vào - Entry criteria: Tập hợp các điều kiện chung và cụ thể.

**Mục đích**: ngăn chặn và loại bỏ các tiêu chí đầu vào lỗi.

* Đánh giá chính thức - formal review : Một đặc tính đánh giá bằng tài liệu và yêu cầu kỹ thuật.
* Đánh giá không chính thức - informal review: Một đánh giá không dựa trên quy trình chính thức.
* Đánh giá kỹ thuật - technical review: Một hoạt động thảo luận nhóm mà tập trung vào việc đạt được sự đồng thuận về các phương pháp kỹ thuật được thực hiện.
* Đánh giá ngang hàng - peer review: Một đánh giá của một sản phẩm phần mềm giữa các đồng nghiệp của nhà sản xuất ra sản phẩm với mục đích xác định các lỗi và cải tiến.
* Thanh tra – inspection: Một loại đánh giá dựa trên thanh tra trực quan tài liệu để phát hiện các lỗi.
* Walkthrough : Nội dung của tài liệu được trình bày từng bước một bởi tác giả của tài liệu để thu thập thông tin và thiết lập hiểu biết chung.
* Người điều hành - moderator: Các trưởng nhóm (leader) và người chịu trách nhiệm chính cho việc thanh tra (inspection) hoặc tiến trình đánh giá khác.
* Người đánh giá - reviewer: Những người tham gia vào việc đánh giá và mô tả sự bất thường trong sản phẩm hoặc dự án được đánh giá. Người đánh giá có thể được chọn để đưa ra các quan điểm khác nhau và vai trò trong quá trình review.
* Người ghi chép: Người ghi lại từng lỗi được đưa ra và bất kỳ đề xuất hoặc thông tin phản hồi được đưa ra trong cuộc họp để cải tiến quy trình.

#### 1.2.2.1. Các hoạt động của đánh giá chính thức (quy trình đánh giá chính thức) - Activities of a Formal Reviews

#### Một đánh giá chính thức điển hình có các hoạt động chính sau đây :

1. Kế hoạch – Planning

* Xác định tiêu chí đánh giá
  + Lựa chọn nhân sự
  + Phân bổ vai trò
  + Xác định tiêu chí đầu vào và tiêu chí kết thúc cho nhiều loại đánh giá chính thức (ví dụ : inspections)

1. Khởi động dự án (Kick-off meeting)
   * Phân phối các tài liệu
   * Giải thích mục tiêu, quy trình ,các tài liệu cho người tham gia
2. Chuẩn bị cá nhân- Individual Preparation
   * Chuẩn bị cho cuộc họp đánh giá bằng cách xem xét các tài liệu.
   * Ghi nhận các lỗi tiềm năng, các câu hỏi và các bình luận.
3. Kiểm tra/ đánh giá/ ghi kết quả ( trong cuộc họp đánh giá) - (Review Meeting)
   * Thảo luận hoặc ghi lại kết quả trong tài liệu trong cuộc họp.
   * Ghi nhận các lỗi, kiến nghị liên quan đến việc xử lý các lỗi, đưa ra quyết định về các lỗi.
   * Kiểm tra/ đánh giá và ghi lại các vấn đề trong cuộc họp.
4. Làm lại - Rework
   * Sửa các lỗi tìm thấy ( thường được thực hiện bởi tác giả).
   * Ghi lại trạng thái đã cập nhật của lỗi.
5. Bước tiếp theo - Follow-up
   * Kiểm tra lại các lỗi đã được giải quyết.
   * Thu thập các số liệu.
   * Kiểm tra tiêu chí kết thúc.

#### 1.2.2.2. Vai trò và trách nhiệm của các thành viên - Roles and Responsibilities

Một đánh giá điển hình sẽ bao gồm:

* Quản lý :
  + Quyết định về việc thực hiện các đánh giá.
  + Phân bổ thời gian trong dự án và xác định các mục tiêu đánh giá đã được đáp ứng hoặc chưa được đáp ứng.
* Người điều hành :
  + Người lãnh đạo xác định loại review, cách tiếp cận và thành phần đội review. Lên lịch trình cho cuộc họp, phổ biến các tài liệu trước khi họp, dẫn dắt vấn đề có thể thảo luận và lưu trữ dữ liệu được thu thập.
  + Dàn xếp giữa các quan điểm khác nhau để đánh giá thành công.
* Tác giả :
  + Người viết hoặc người có trách nhiệm chính đối với các tài liệu được đánh giá.
* Người đánh giá:
  + tham gia rà soát, kiểm sai sót 🡪 quan trọng nhất
  + tìm kiếm sai sót.
* Người ghi chép :
  + Ghi chép tài liệu về tất cả các vấn đề và các quan điểm mở được xác định trong cuộc họp đánh giá.

#### 1.2.2.3. Các loại đánh giá - Types of Reviews

Walk-through, đánh giá không chính thức (informal review), đánh giá kĩ thuật (technical review) và thanh tra (inspection) là các kĩ thuật cơ bản được sử dụng trong suốt quá trình phát triển phần mềm.

Các đặc điểm và mục đích chính của các loại review:

* Đánh giá không chính thức (Informal Review)
  + Không có quy trình chính thức, trao đổi ngoài lề.
  + Có thể lấy form của các chương trình hoặc của một trưởng nhóm kỹ thuật đánh giá thiết kế.
  + Kết quả có thể được ghi lại.
  + Thay đổi lợi ích phụ thuộc vào những người đánh giá.
  + Lập trình theo cặp
  + Mục đích chính:
    - Chi phí thấp để có được một số lợi ích.
    - Tìm các lỗi.
* Walkthrough: là một cuộc họp để đánh giá, thường là không có sự chuẩn bị trước, có thể xảy ra bất cứ lúc nào và không có lịch trình.
  + Không phải là một quy trình đánh giá chính thức (semi-formal).
  + Chủ trì bởi tác giả của tài liệu, mã nguồn.
* Tài liệu, mã nguồn được giải thích từng bước nhằm :
* Đạt sự hiểu rõ, đồng thuận
* Thu thập thông tin, phản hồi(thảo luận, đề xuất, góp ý)
* Người tham gia:
* Trong hoặc ngoài nhóm 🡪 đề xuất sửa chữa tốt hơn
* Trong hoặc ngoài ngành 🡪 nhiều ý kiến hơn, hỗ trợ lẫn nhau.
  + Mục đích chính :
    - Thu thập thông tin và đánh giá nội dung tài liệu
    - Đạt được sự hiểu biết chung.
    - Thảo luận về các giải pháp đề xuất.
* Đánh giá kỹ thuật
  + Quy trình đánh giá chính thức.
  + Lập tài liệu, xác định quá trình phát hiện những sai sót bao gồm người điều hành, các đồng nghiệp, các chuyên gia kỹ thuật.
  + Ý tưởng được dẫn dắt bởi người điều hành( không phải tác giả)
  + Tùy chọn sử dụng các danh sách kiểm tra (checklists)
  + Chuẩn bị báo cáo đánh giá bao gồm danh sách các phát hiện, các quyết định xem sản phẩm phần mềm có đáp ứng được các yêu cầu, kiến nghị liên quan đến các phát hiện.
  + Mục đích chính:
    - Thảo luận
    - Đưa ra các quyết định
    - Đánh giá các lựa chọn thay thế
    - Tìm các lỗi
    - Giải quyết các vấn đề kỹ thuật
    - Kiểm tra sự phù hợp với các thông số kỹ thuật, kế hoạch, quy định và tiêu chuẩn.
* Inspection – thanh tra:
  + Là loại đánh giá hình thức nhất -> theo quy trình các bước.
  + Chủ trì bởi người điều hành đã được đào tạo( không phải tác giả).
  + Tiến hành như một đánh giá ngang hàng ( đánh giá giữa các đồng nghiệp)
  + Tài liệu/ mã nguồn được rà soát theo các nguyên tắc(rules) và danh sách cần kiểm tra (checklist). Tất cả các sai sót được lập văn bản.
  + Người tham gia : phân rõ vai trò của người tham gia trong quy trình.
  + Quá trình đánh giá chính thức dựa trên các nguyên tắc và danh sách kiểm tra
  + Mục đích chính :
    - Phát hiện các lỗi

#### 1.2.2.4. Các yếu tố thành công cho đánh giá - Success Factors for Reviews

Các yếu tố thành công cho đánh giá bao gồm:

* Mỗi đánh giá có mục tiêu rõ ràng được xác định trước.
* Những người tham gia phù hợp với các mục tiêu đánh giá.
* Người kiểm thử là người đánh giá có giá trị đóng góp vào việc đánh giá và cũng tìm hiểu về sản phẩm để họ chuẩn bị cho các kiểm thử sớm.
* Các lỗi tìm thấy được tiếp nhận và bày tỏ quan điểm.
* Vấn đề con người và khía cạnh tâm lý được giải quyết.
* Đánh giá được tiến hành trong không khí tin tưởng, kết quả sẽ không được sử dụng cho việc đánh giá những người tham gia.
* Danh sách kiểm tra(checklist) hoặc các vai trò được sử dụng phù hợp để tăng hiệu quả của việc xác định các lỗi.
* Tầm quan trọng trong việc học tập và cải tiến quá trình.

**Phân tích tĩnh bằng công cụ (static analysis)**

Các định nghĩa :

* Trình biên dịch - Compiler : Một công cụ phần mềm biên dịch các chương trình bằng một ngôn ngữ bậc cao thành ngôn ngữ máy.
* Sự phức tạp - complexity : Mức độ mà một thành phần hoặc hệ thống có một thiết kế hoặc cấu trúc bên trong khó hiểu
* Luồng điều khiển - control flow : Một chuỗi các sự kiện( đường dẫn) trong việc thực hiện thông qua một thành phần hoặc hệ thống.
* Luồng dữ liệu - data flow : Một đại diện của các trình tự có thể thay đổi trạng thái của đối tượng dữ liệu, trạng thái của đối tượng có thể là sự tạo thành(creation), cách sử dụng (usage), sự tiêu hủy (destruction).
* Phân tích tĩnh - static analysis : Phân tích hiện vật phần mềm (ví dụ như yêu cầu của code), thực hiện mà không cần thực thi các hiện vậtt phát triển phần mềm.

Mục tiêu của phân tích tĩnh là tìm ra các khiếm khuyết trong mã nguồn (source code) phần mềm và trong mô hình phần mềm. Phân tích tĩnh được thực hiện mà không cần chạy phần mềm, được kiểm tra bằng công cụ. Công cụ phân tích tĩnh phân tích mã chương trình (ví dụ : luồng điều khiển và luồng dữ liệu).

* Lợi ích của phân tích tĩnh:
  + Phát hiện sớm các lỗi trước khi thực hiện kiểm thử
  + Cảnh báo sớm về những khía cạnh đáng ngờ của code hoặc thiết kế bằng cách tính toán các số liệu.
  + Xác định các lỗi không dễ dàng tìm được trong kiểm thử động.
  + Xác định phần phụ thuộc và không nhất quán trong mô hình phần mềm.
  + Cải thiện khả năng bảo trì của code và thiết kế
  + Ngăn ngừa các lỗi , cải thiện chất lượng.
* Các khiếm khuyết điển hình được phát hiện bằng công cụ phân tích tĩnh bao gồm :
  + Tham chiếu một biến với một giá trị không xác định
  + Giao diện không đồng nhất giữa các modules và các thành phần
  + Các biến không được sử dụng hoặc khai báo không đúng
  + Làm chết code
  + Logic thiếu và sai( vòng lặp vô hạn)
  + Cấu trúc quá phức tạp
  + Vi phạm tiêu chuẩn chương trình
  + Lỗ hổng bảo mật
  + Vi phạm cú pháp của mã (code) và mô hình phần mềm
* Phân tích tĩnh bằng công cụ khi nào và do ai thực hiện:
  + Sử dụng bởi các nhà phát triển trước và trong kiểm thử thành phần và kiểm thử tích hợp
  + Khi kiểm tra trong code bằng công cụ quản lý cấu hình và thực hiện bởi người phát triển
  + Trong mô hình phần mềm thực hiện bởi người thiết kế.
* Phân tích tĩnh bằng công cụ có thể tạo ra một số lượng lớn các tin nhắn cảnh báo, cần quản lý tốt để sử dụng công cụ một cách hiệu quả nhất.
* Trình biên dịch có thể hỗ trợ phân tích tĩnh, bao gồm cả tính toán các số liệu.