**Unit Test**

1. **Unit test là gì?**

* Unit test là một loại kiểm thử phần mềm trong lĩnh vực phát triển phần mềm. Trong kiểm thử này, các đơn vị nhỏ nhất của mã nguồn, gọi là "đơn vị" (unit), được kiểm tra để đảm bảo rằng chúng hoạt động như mong đợi và không có lỗi.
* Mỗi "đơn vị" thường là một hàm, một phương thức, hoặc thậm chí là một dòng code.
* Mục tiêu của unit test là cung cấp một phản hồi nhanh chóng và tự động về tính đúng đắn của từng đơn vị mã nguồn. Bằng cách này, nó giúp đảm bảo rằng các thay đổi trong mã nguồn không làm hỏng các phần khác, và giúp giảm thiểu lỗi khi phát triển phần mềm.

1. **Tại sao cần kiểm thử phần mềm?**

* **Đảm bảo tính đáng tin cậy**: Đảm bảo rằng phần mềm hoạt động như mong đợi và không gây ra các lỗi không mong muốn.
* **Phát hiện lỗi sớm:** Tránh được việc lỗi trở nên phức tạp và tốn kém để sửa chữa sau này.
* **Giảm thiểu rủi ro:** Việc không kiểm thử có thể dẫn đến những lỗi không mong muốn xuất hiện trong sản phẩm cuối cùng, làm giảm uy tín của sản phẩm và gây tổn thất kinh tế cho doanh nghiệp.
* **Tăng sự linh hoạt:** Có một bộ kiểm thử mạnh mẽ giúp dễ dàng thay đổi và mở rộng phần mềm mà không lo sợ gây ra các vấn đề không mong muốn trong các tính năng hiện tại.
* **Tăng sự hiểu biết về sản phẩm:** Giúp các nhà phát triển hiểu rõ hơn về cách hoạt động của phần mềm và yêu cầu của người dùng.
* **Đáp ứng nhanh chóng với thay đổi:** Các bộ kiểm thử tự động giúp đảm bảo rằng các tính năng mới hoặc thay đổi sẽ không làm hỏng các tính năng hiện tại.

1. **Quy trình kiểm thử**

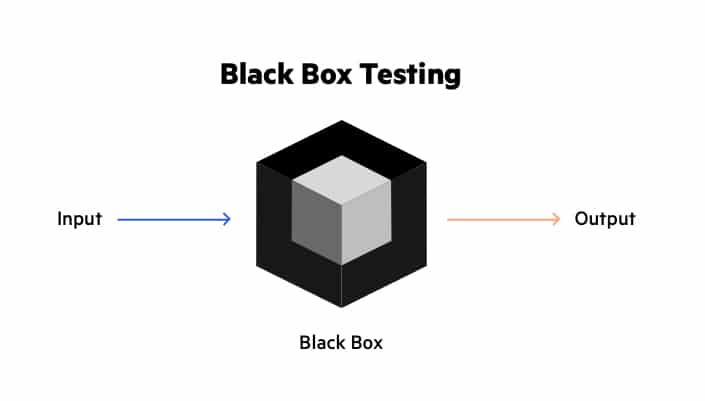
* **Xác định các đơn vị kiểm thử:** Các đơn vị này thường là các hàm, phương thức hoặc thậm chí là các dòng code cụ thể.
* **Thiết kế các test case:** Viết các test case để kiểm tra các đơn vị đã xác định. Mỗi test case nên kiểm tra một khía cạnh cụ thể của đơn vị, ví dụ: trường hợp đầu vào hợp lệ, trường hợp đầu vào không hợp lệ, v.v.
* **Chạy các test case:** Chạy các test case để kiểm tra các đơn vị đã xác định. Đảm bảo rằng tất cả các test case đều chạy mà không có lỗi và cho kết quả mong đợi.
* **Xem xét và sửa lỗi (nếu cần):** Nếu có bất kỳ test case nào không thành công, xem xét lỗi và sửa chữa chúng trong mã nguồn. Sau đó, chạy lại các test case để đảm bảo rằng lỗi đã được khắc phục.

1. **Kỹ thuật unit testing**

* **Được chia thành 2 loại:** kiểm thử hộp đen và kiểm thử hộp trắng.
* **Kiểm thử hộp đen:**
* Khái niệm:
* Là một phương pháp kiểm thử phần mềm được thực hiện mà không biết được cấu tạo bên trong của phần mềm
* Các tester kiểm tra xem hệ thống như một chiếc hộp đen, không có cách nào nhìn thấy bên trong của cái hộp.
* Phương pháp: Function test và System test

Phương pháp này cố tìm các lỗi trong các loại sau:

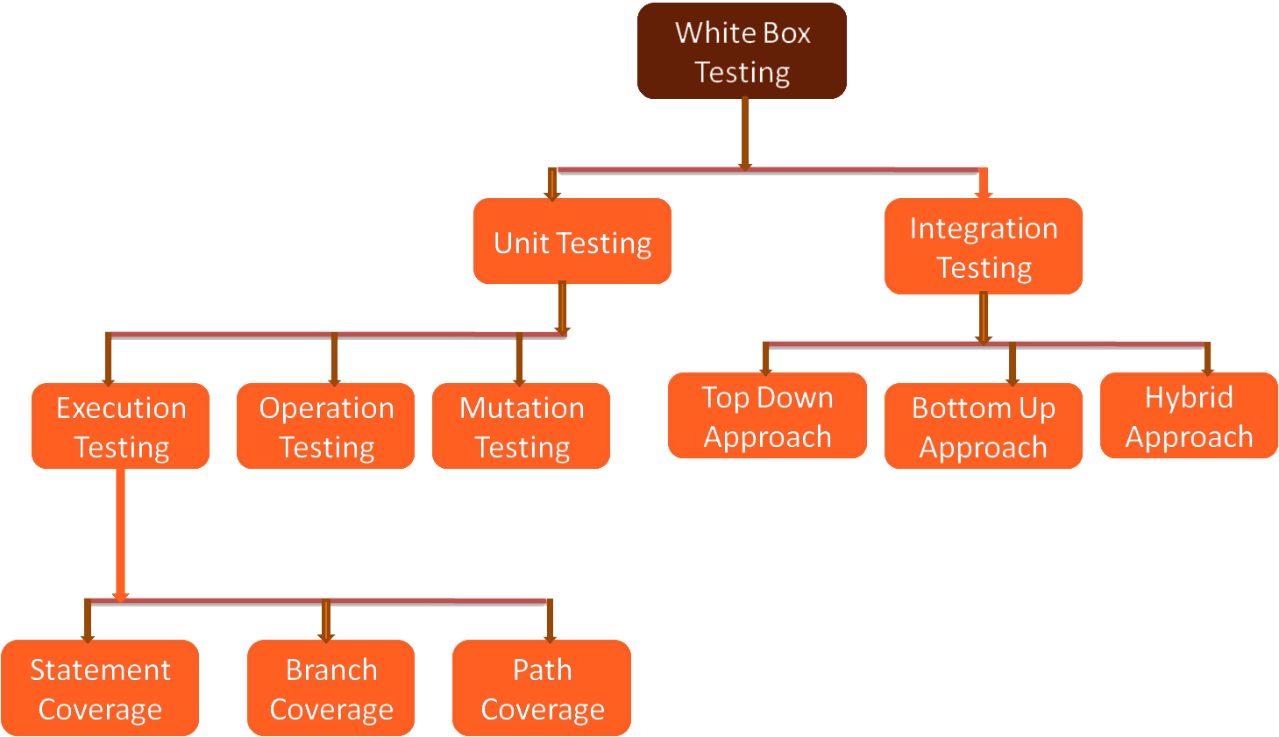
* Chức năng không chính xác hoặc thiếu.
* Lỗi giao diện.
* Lỗi trong cấu trúc dữ liệu hoặc truy cập cơ sở dữ liệu bên ngoài.
* Hành vi hoặc hiệu suất lỗi.
* Khởi tạo và chấm dứt các lỗi.



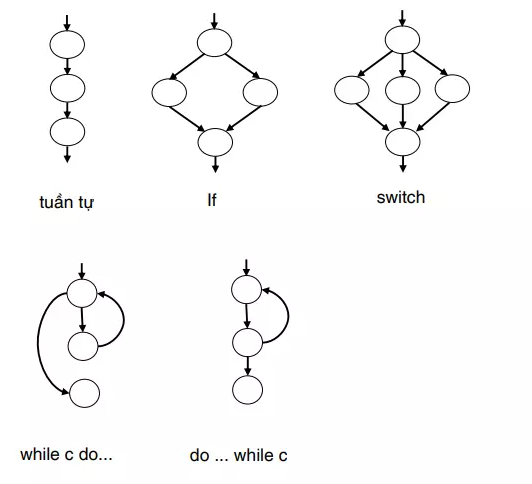
* Quy trình kiểm thử hộp đen:
* Lập quy trình test:



* Thiết kế test case.
* Thực hiện kiểm thử.
* Báo cáo kiểm thử.
* Ưu điểm của kiểm thử hộp đen:
* Kiểm thử hộp đen giúp phát hiện lỗi từ góc nhìn của người dùng, giúp cải thiện chất lượng của phần mềm.
* Các tester không có “mối ràng buộc” nào với code, tester black box tìm được nhiều bug ở nơi mà các developer không tìm thấy.
* Do không cần biết về cấu trúc mã nguồn, các kịch bản kiểm thử hộp đen thường dễ dàng thực hiện tự động, giúp tiết kiệm thời gian và công sức.
* Các tester có thể được thực hiện bởi một cơ quan độc lập từ các developer, cho phép một cái nhìn khách quan và tránh sự phát triển thiên vị.
* Thường kiểm tra các giao diện và trải nghiệm người dùng, giúp đảm bảo rằng phần mềm đáp ứng được mong đợi của người dùng cuối.
* **Kiểm thử hộp trắng:**
* Khái niệm:
* 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.
* Phương pháp:
* Unit Testing (Kiểm thử đơn vị): Để kiểm tra đường dẫn trong một đơn vị.
* Integration Testing (Test tích hợp): Để kiểm tra đường dẫn giữa các đơn vị.
* System Testing (Test hệ thống): Để kiểm tra các đường dẫn giữa các hệ thống con.



* Các kỹ thuật kiểm thử hộp trắng phổ biến:
* Kiểm thử cơ bản – Đồ thị dòng
* Kiểm thử nhánh
* Kiểm thử đường dẫn
* Kiểm thử vòng lặp



Một vài kiểu cấu trúc thành phần đồ thị dòng

* Ưu điểm của kiểm thử hộp trắng:
* Test có thể bắt đầu ở giai đoạn sớm hơn, không cần phải chờ đợi GUI.
* Test kỹ càng hơn, có thể bao phủ hầu hết các đường dẫn.
* Thích hợp trong việc tìm kiếm lỗi và các vấn đề trong mã lệnh.
* Cho phép tìm kiếm các lỗi ẩn bên trong.
* Các lập trình viên có thể tự kiểm tra.
* Giúp tối ưu việc mã hoá.

1. **Ứng dụng unit test vào ứng dụng quản lý điểm & theo dõi học tập**

* Áp dụng unit test trong quá trình phát triển để kiểm tra tính đúng đắn của các tính năng của ứng dụng:
* Tính năng nhập điểm
* Tính năng tính điểm trung bình
* Tính năng phân loại, xếp loại học lực
* Tính năng xuất bảng điểm
* Tính năng xử lý ngoại lệ
* Bằng cách áp dụng unit test vào các tính năng của ứng dụng, ta có thể đảm bảo tính đúng đắn và hiệu quả của ứng dụng, giúp tăng cường niềm tin của người dùng và giảm thiểu rủi ro lỗi trong quá trình sử dụng.

1. **Kết luận**

* Unit test là một phần quan trọng của quy trình phát triển phần mềm, giúp đảm bảo tính đáng tin cậy, chất lượng và khả năng mở rộng của mã nguồn. Bằng cách kiểm tra từng đơn vị mã nguồn một cách tự động và liên tục, unit test giúp phát hiện và sửa chữa lỗi một cách nhanh chóng, tăng cường sự tự tin trong việc triển khai và duy trì sản phẩm phần mềm.