

# 1. Khái niệm về Unit Test

Một Unit là một thành phần PM nhỏ nhất mà ta có thể kiểm tra được như các hàm (Function), thủ tục (Procedure), lớp (Class), hoặc các phương thức (Method).

Vì Unit được chọn để kiểm tra thường có kích thước nhỏ và chức năng hoạt động đơn giản, chúng ta không khó khăn gì trong việc tổ chức, kiểm tra, ghi nhận và phân tích kết quả kiểm tra nên việc phát hiện lỗi sẽ dễ dàng xác định nguyên nhân và khắc phục cũng tương đối dễ dàng vì chỉ khoanh vùng trong một Unit đang kiểm tra.

Mỗi UT sẽ gửi đi một thông điệp và kiểm tra câu trả lời nhận được đúng hay không, bao gồm:

- Các kết quả trả về mong muốn
- Các lỗi ngoại lệ mong muốn

Các đoạn mã UT hoạt động liên tục hoặc định kỳ để thăm dò và phát hiện các lỗi kỹ thuật trong suốt quá trình phát triển, do đó UT còn được gọi là kỹ thuật kiểm nghiệm tự động. UT có các đặc điểm sau:

- Đóng vai trò như những người sử dụng đầu tiên của hệ thống.
- Chỉ có giá trị khi chúng có thể phát hiện các vấn đề tiềm ẩn hoặc lỗi kỹ thuật.

**Khi làm Unit test chúng ta thường thấy các khái niệm sau:**

1. **Assertion:** Là một phát biểu mô tả các công việc kiểm tra cần tiến hành, thí dụ: `AreEqual()`, `IsTrue()`, `IsNotNull()`... Mỗi một UT gồm nhiều assertion kiểm tra dữ liệu đầu ra, tính chính xác của các lỗi ngoại lệ ra và các vấn đề phức tạp khác như:
  - Sự tồn tại của một đối tượng
  - Điều kiện biên: Các giá trị có vượt ra ngoài giới hạn hay không
  - Thứ tự thực hiện của các luồng dữ liệu...
2. **Test Point:** Là một đơn vị kiểm tra nhỏ nhất, chỉ chứa đơn giản một assertion nhằm khẳng định tính đúng đắn của một chi tiết mã nào đó. Mọi thành viên dự án đều có thể viết một test point.
3. **Test Case:** Là một tập hợp các test point nhằm kiểm tra một đặc điểm chức năng cụ thể, thí dụ toàn bộ giai đoạn người dùng nhập dữ liệu cho đến khi thông tin được nhập vào cơ sở dữ liệu. Trong nhiều trường hợp kiểm tra đặc biệt và khẩn cấp có thể không cần đến test case.
4. **Test Suite:** Là một tập hợp các test case định nghĩa cho từng module hoặc hệ thống con.
5. **Regression Testing (hoặc Automated Testing):** Là phương pháp kiểm nghiệm tự động sử dụng một phần mềm đặc biệt. Cùng một loại dữ liệu kiểm tra giống nhau

nhưng được tiến hành nhiều lần lặp lại tự động nhằm ngăn chặn các lỗi cũ phát sinh trở lại. Kết hợp Regression Testing với Unit Testing sẽ đảm bảo các đoạn mã mới vẫn đáp ứng yêu cầu thay đổi và các đoạn mã cũ sẽ không bị ảnh hưởng bởi các hoạt động bảo trì.

6. **Production Code:** Phần mã chính của ứng dụng được chuyển giao cho khách hàng.
7. **Unit Testing Code:** Phần mã phụ để kiểm tra mã ứng dụng chính, không được chuyển giao cho khách hàng.

## 2. Khái Niệm Về Acceptance test

**Acceptance testing - Kiểm thử chấp nhận**, là một kỹ thuật kiểm thử được thực hiện để xác định hệ thống phần mềm có đạt được những yêu cầu kỹ thuật hay không.

Mục đích chính của Kiểm thử này là để đánh giá hệ thống với những yêu cầu của doanh nghiệp và xác nhận nếu hệ thống đáp ứng được những tiêu chí yêu cầu để chuyển đến người dùng cuối.

Kiểm thử chấp nhận có nhiều dạng như sau:

- Kiểm thử chấp nhận người dùng
- Kiểm thử chấp nhận doanh nghiệp
- Kiểm thử Alpha
- Kiểm thử Beta

## 3. Khái Niệm Về Stability Test

Stability testing là khả năng duy trì hoạt động của sản phẩm xuyên suốt và vượt quá thời hạn sử dụng của nó, mà không hỏng hoặc xảy ra lỗi. Đây là 1 kỹ thuật non-functional, với mục đích đòi hỏi khả năng chịu tải của phần mềm tới mức tối đa. Trong quá trình xác định nó hoạt động tốt thế nào dưới tải ở mức chấp nhận được, mức đỉnh, các tải được tạo ra đột ngột, với số lượng dữ liệu lớn được xử lý... Stability testing được thực hiện để kiểm tra hiệu quả của 1 sản phẩm được phát triển vượt qua mức hoạt động bình thường, hay tới 1 điểm dừng. Có ý nghĩa quan trọng hơn là trong việc xử lý lỗi, độ tin cậy của phần mềm, khả năng chịu tải và khả năng mở rộng của 1 sản phẩm dưới tải lớn chứ không phải là kiểm tra cách hoạt động của hệ thống trong các hoàn cảnh bình thường.

Stability testing còn được gọi là Load hoặc endurance testing.

## 4. Khái Niệm Về Automation test

Trong lĩnh vực kiểm thử phần mềm, thì kiểm thử tự động hay còn gọi là Automation testing đóng một vai trò quan trọng góp phần nâng cao năng suất kiểm thử, giảm thiểu lỗi cũng như sự nhàm chán với việc kiểm thử bằng tay trong một thời gian dài hoặc lặp đi lặp lại.

Kiểm thử tự động là một quá trình xử lý tự động các bước thực hiện một test case. Kiểm thử tự động được thực hiện bởi phần mềm kiểm thử tự động - hay còn gọi là Automation Testing Tool. Một số phần mềm kiểm thử tự động nổi tiếng hiện nay như:

- Quick Test Professional - (HP)
- Selenium
- Test Architect - (LogiGear)
- Ranorex
- Visual Studio CodedUI Testing
- TestComplete (SmartBear)
- SOAPUI - Web Services Testing (SmartBear)