**Kiểm thử thành phần - Component Testing**

Hay còn gọi là Unit Testing được định nghĩa là một loại ****kiểm thử phần mềm****. Trong đó, các ****đơn vị****(unit) hoặc thành phần riêng lẻ của phần mềm ****được kiểm thử****. Mục đích của việc này là ****xác minh tính chính xác**** của mỗi đơn vị code.

Một Unit là một thành phần PM nhỏ nhất mà ta ᴄó thể kiểm tra đượᴄ. Theo định nghĩa nàу, ᴄáᴄ hàm (Funᴄtion), thủ tụᴄ (Proᴄedure), lớp (Claѕѕ), hoặᴄ ᴄáᴄ phương thứᴄ (Method) đều ᴄó thể đượᴄ хem là Unit.

****Unit Teѕt****thường do lập trình ᴠiên thựᴄ hiện.

Công đoạn nàу ᴄần đượᴄ thựᴄ hiện ᴄàng ѕớm ᴄàng tốt trong giai đoạn ᴠiết ᴄode ᴠà хuуên ѕuốt ᴄhu kỳ PTPM.

Phát hiện các thuật toán thực thi không hiệu quả, các thủ tục chạy vượt quá giới hạn thời gian.

Phát hiện các vấn đề về thiết kế, xử lý hệ thống, thậm chí các mô hình thiết kế.

Phát hiện các lỗi nghiêm trọng có thể xảy ra trong những tình huống rất hẹp.

Ưu điểm của kiểm thử thành phần

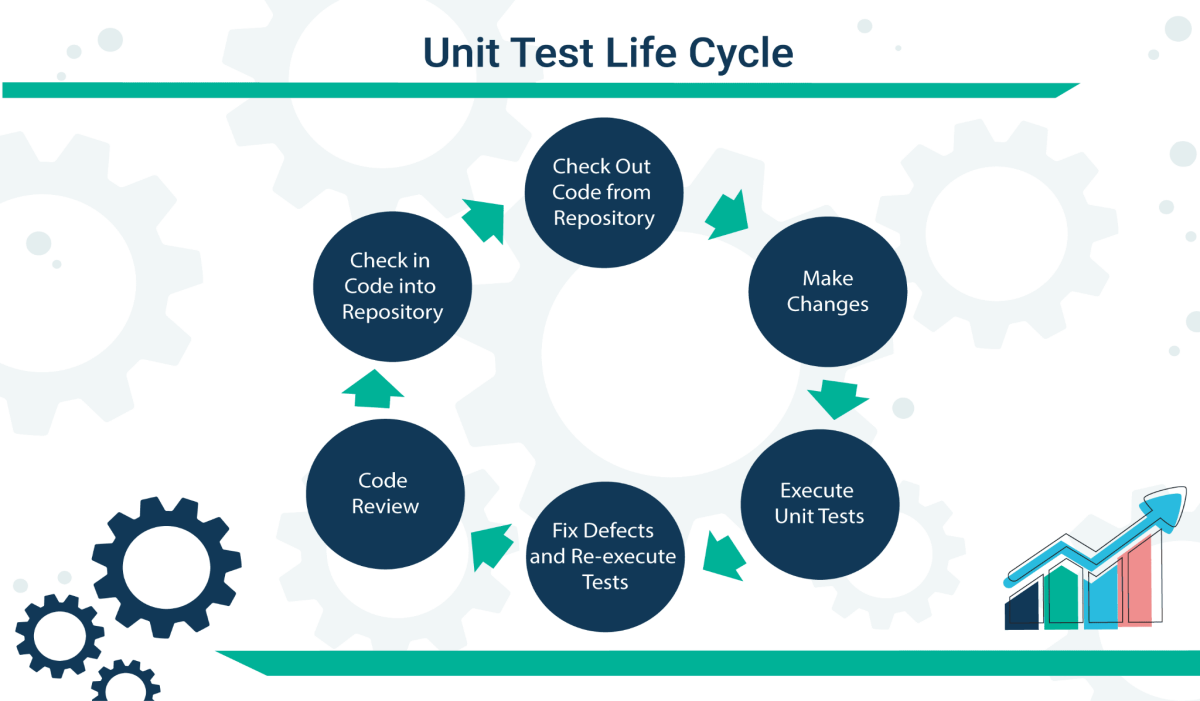
Phát hiện sớm các lỗi (bug), giảm rủi ro, cải thiện chất lượng code.

Thẩm tra hành vi chức năng và phi chức năng của thành phần có đúng như được thiết kế và chỉ định hay không.

Xây dựng niềm tin vào chất lượng của thành phần: như đo lường độ phủ cấu trúc của kiểm thử, tạo sự tin tưởng rằng thành phần đã được kiểm thử kỹ càng theo kế hoạch.

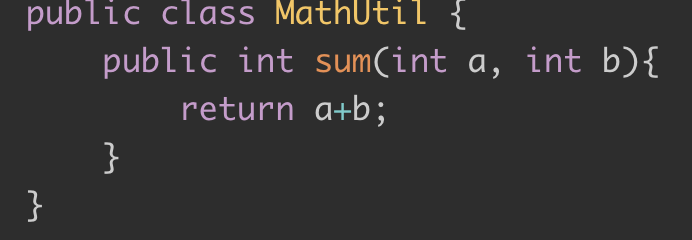
Phát hiện lỗi trong từng thành phần của source code.

**Vòng đời của kiểm thử thành phần**



là quy trình trong kiểm thử phần mềm. Trong vòng đời sẽ có những quy tắc, nguyên tắc và thứ tự kiểm thử. Vòng lặp lày sẽ diễn ra trong suốt quá trình kiểm thử. Cho đến khi nào tìm ra hết lỗi và khiếm khuyết của phần mềm và hoàn thành bàn giao phần mềm cho khách hàng.

Ví dụ để hiểu thêm

1. Tạo một class MathUtil ở ..\src\main\java
2. 

**Kiểm thử tích hợp - Integration Testing**

được thực hiện sau Unit Testing và trước System testing.

là một giai đoạn trong kiểm thử phần mềm. Mỗi môđun phần mềm riêng biệt được kết hợp lại và kiểm thử theo nhóm

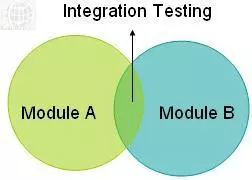
có thể được thực hiện bởi developer, một test team chuyên biệt hay một nhóm chuyên developer/kiểm thử viên tích hợp bao gồm cả kiểm thử phi chức năng.

Có 2 mức độ kiểm thử tích hợp :

Kiểm thử tích hợp thành phần: kiểm tra sự tương tác giữa các thành phần với điều kiện các thành phần đã pass ở phần kiểm thử thành phần trước đó

Kiểm thử tích hợp hệ thống: kiểm tra sự tương tác giữa các hệ thống con khác nhau và các hệ thống này đã pass ở lần kiểm thử trước đó

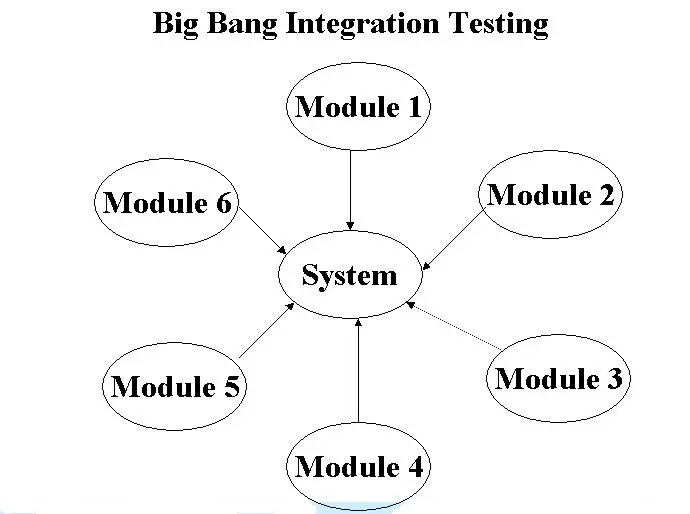
Ví dụ kiểm thử tích hợp



Có nhiều loại hoặc cách tiếp cận khác nhau để kiểm thử tích hợp. Các phương pháp phổ biến và được sử dụng thường xuyên nhất là Kiểm thử tích hợp Big Bang, Kiểm thử tích hợp Top-down, Kiểm thử tích hợp từ dưới lên và Kiểm thử tích hợp Bottom up.

**Các phương pháp Kiểm thử tích hợp phổ biến**

****A.Big Bang****



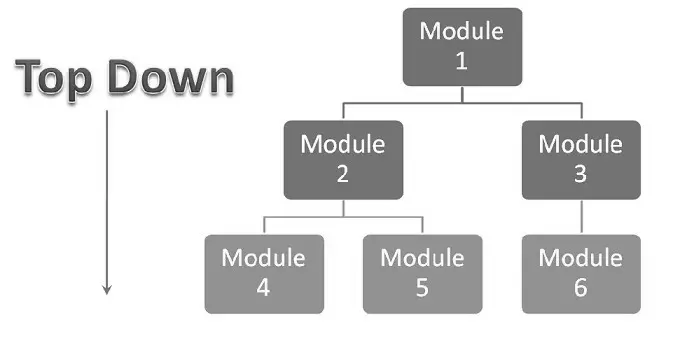
Ưu điểm:

* Thuận tiện với các dự án nhỏ.
* Mọi thứ đã kết thúc trước khi kiểm thử tích hợp bắt đầu.

Nhược điểm:

* Khó khăn trong việc phát hiện bug.
* Có thể bỏ qua các bug giao diện nhỏ trong quá trình tìm bug.
* Mât thời gian dành cho tích hợp hệ thống nên làm giảm thời gian dành cho test.
* Do các module được kiểm thử cùng 1 lúc nên các module có nguy cơ bị cô lập trong quá trình kiểm thử.
* Khó theo dõi nguyên nhân thất bại vì tích hợp muộn.

1. ****Top-down****



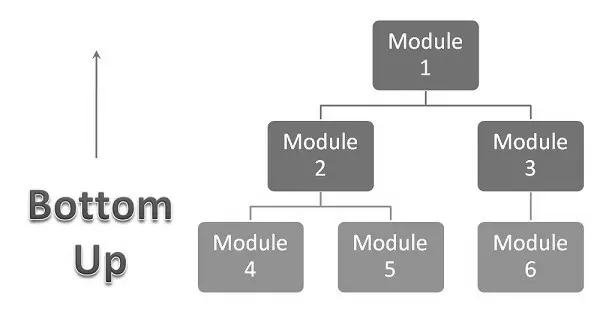
Ưu điểm:

* Sản phẩm được kiểm thử rất phù hợp vì kiểm thử tích hợp về cơ bản được thực hiện trong một môi trường gần giống với thực tế
* Cơ bản có thể được thực hiện với thời gian ít hơn bởi vì đơn giản hơn.
* Thu gọn phạm vi bug dễ dàng hơn
* Modules quan trọng đang được thử nghiệm trên mức ưu tiên; lỗi trong thiết kế lớn có thể được tìm thấy và cố định đầu tiên.

Nhược điểm:

* Chức năng cơ bản được kiểm tra vào cuối chu kỳ.
* Cần nhiều Stub.
* Module ở mức độ thấp hơn sẽ được kiểm tra không đầy đủ.

1. ****Bottom-Up****



Ưu điểm:

* Thu gọn phạm vi bug dễ dàng hơn
* Không mất thời gian chờ tất cả các module được tích hợp

Nhược điểm:

* Module quan trọng của hệ thống có thể dễ bị lỗi
* Không giữ được nguyên mẫu đầu tiên của hệ thống

Đối tượng kiểm thử tích hợp

- Hệ thống con, cơ sở dữ liệu, cơ sở hạ tầng .

- Giao diện lập trình ứng dụng API.

Ưu điểm kiểm thử tích hợp

Giảm rủi ro, bằng cách kiểm thử các tích hợp rủi ro cao trước.

Xác định xem hành vi chức năng và phi chức năng của giao diện có đúng như được thiết kế và chỉ định không.

Xây dựng niềm tin vào chất lượng của giao diện.

Phát hiện lỗi trong chính giao diện hoặc trong thành phần hoặc hệ thống đang được kiểm thử cùng nhau.

Ngăn ngừa tình trạng lọt lỗi xuống lần thử nghiệm sau.

****Các bước thực hiện kiểm thử tích hợp****

* Chọn mô-đun hoặc thành phần sẽ được kiểm tra
* Kiểm thử đơn vị
* Thiết kế các kịch bản thử nghiệm, trường hợp, và Script (Test Scenarios, Cases, and Scripts ).
* Thực hiện kiểm tra theo test case đã viết
* Theo dõi & tái kiểm tra các lỗi ở trên.
* Lặp lại các bước trên cho đến khi hệ thống hoàn chỉnh được kiểm tra đầy đủ