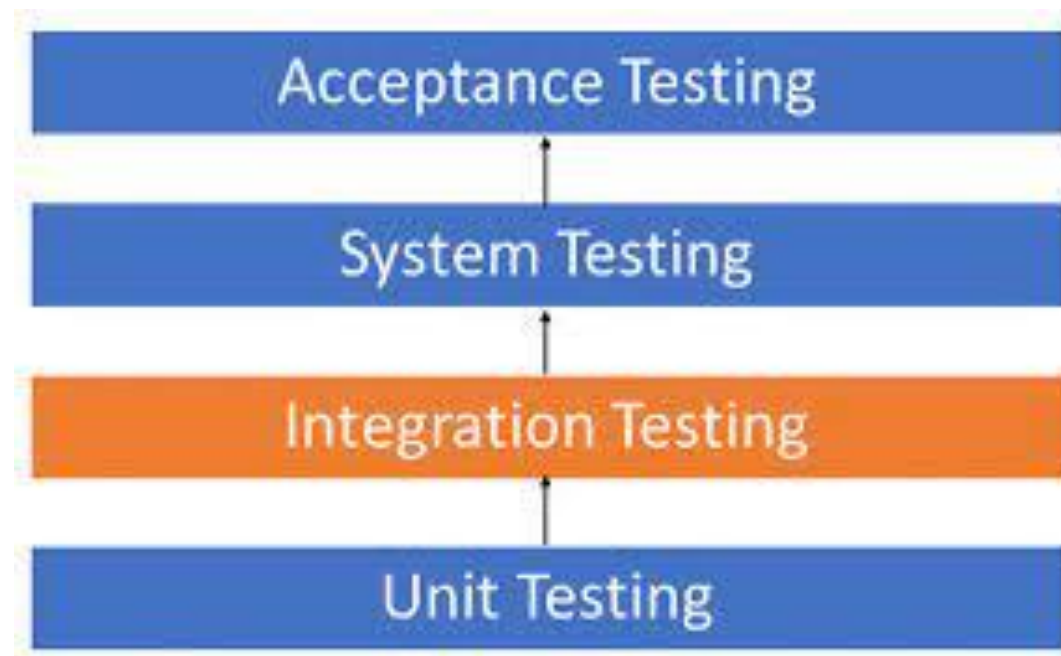


Tìm hiểu về INTEGRATED TEST

Khái niệm test tích hợp

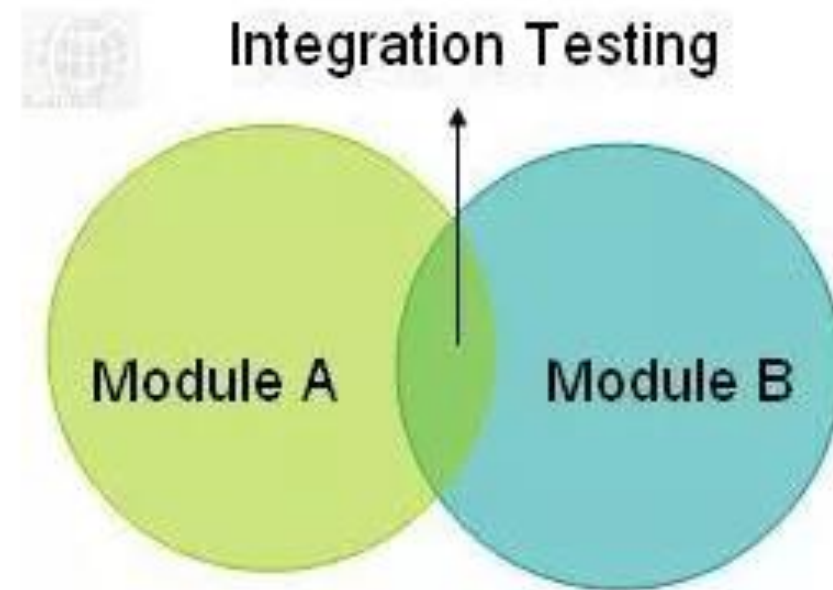
- Kiểm thử tích hợp (Integration testing) hay còn gọi là tích hợp và kiểm thử (integration and testing, viết tắt: I&T) là một giai đoạn trong kiểm thử phần mềm. Mỗi môđun phần mềm riêng biệt được kết hợp lại và kiểm thử theo nhóm.
- Kiểm thử tích hợp xảy ra sau kiểm thử đơn vị (Unit Test) và trước kiểm thử xác nhận. Kiểm thử tích hợp nhận các môđun đầu vào đã được kiểm thử đơn vị, nhóm chúng vào các tập hợp lớn hơn, áp dụng các ca kiểm thử đã được định nghĩa trong kế hoạch kiểm thử tích hợp vào tập hợp đó, và cung cấp đầu ra cho hệ thống tích hợp.



Tại sao lại phải thực hiện kiểm thử tích hợp?

Mặc dù mỗi module đều được kiểm thử đơn vị (Unit test) nhưng các lỗi vẫn còn tồn tại với các nguyên nhân sau:

- Một Module nói chung được thiết kế bởi một lập trình viên có hiểu biết và logic lập trình có thể khác với các lập trình viên khác. Kiểm thử tích hợp là cần thiết để đảm bảo tính hợp nhất của phần mềm.
- Tại thời điểm phát triển module vẫn có thể có thay đổi trong spec của khách hàng, những thay đổi này có thể không được kiểm tra ở giai đoạn unit test trước đó.
- Giao diện và cơ sở dữ liệu của các module có thể chưa hoàn chỉnh khi được ghép lại.
- Khi tích hợp hệ thống các module có thể không tương thích với cấu hình chung của hệ thống.
- Thiếu các xử lý ngoại lệ có thể xảy ra.




Ví dụ về Kiểm thử Tích Hợp

Giả sử bạn làm việc cho một tổ chức CNTT đã được yêu cầu phát triển trang web mua sắm trực tuyến cho Camp World, một công ty bán dụng cụ cắm trại. Sau khi thu thập yêu cầu, phân tích và thiết kế hoàn tất, một nhà phát triển đã được chỉ định để phát triển từng mô-đun bên dưới.

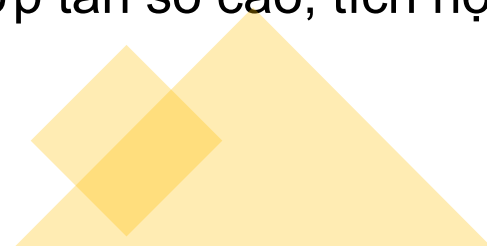
- ❖ Đăng ký và xác thực người dùng / Đăng nhập
- ❖ Danh mục sản phẩm
- ❖ Giỏ hàng
- ❖ Thanh toán
- ❖ Tích hợp cổng thanh toán
- ❖ Theo dõi vận chuyển và gói hàng

Ví dụ về Kiểm thử Tích Hợp

- ❖ Sau khi mỗi mô-đun được gán cho nhà phát triển, nhà phát triển bắt đầu mã hóa chức năng trên các máy riêng lẻ của họ.
- ❖ Họ đã triển khai các mô-đun tương ứng trên các máy của mình để xem những gì đã hoạt động và những gì đã làm, khi họ bắt đầu phát triển mô-đun.
- ❖ Sau khi họ hoàn thành việc phát triển, các nhà phát triển đã kiểm tra các chức năng cá nhân của họ như là một phần của kiểm thử đơn vị của họ và tìm thấy một số khiếm khuyết. Họ đã sửa những khuyết điểm này.
- ❖ Tại thời điểm này, họ cảm thấy các mô-đun của họ đã hoàn thành.
- ❖ Kiểm tra tích hợp nên được thực hiện để xác nhận rằng tất cả các mô-đun hoạt động cùng nhau.
- ❖ Khi họ triển khai tất cả mã của họ trong một máy chung, họ thấy rằng ứng dụng không hoạt động như mong đợi vì các mô-đun riêng lẻ không hoạt động tốt với nhau.
- ❖ Có một số lỗi như - sau khi đăng nhập, giỏ hàng của người dùng không hiển thị các mục họ đã thêm trước đó, số tiền hóa đơn không bao gồm chi phí vận chuyển, v.v.
- ❖ Theo cách này, Kiểm thử tích hợp giúp chúng ta xác định, khắc phục các sự cố và đảm bảo rằng toàn bộ ứng dụng hoạt động như mong đợi.



Cách tiếp cận, phương pháp, chiến lược của 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.
 - ❖ Sự lựa chọn của phương pháp phụ thuộc vào các yếu tố khác nhau như chi phí, độ phức tạp, mức độ quan trọng của ứng dụng, v.v.
 - ❖ Ngoài ra, có nhiều loại thử nghiệm tích hợp ít được biết đến như tích hợp dịch vụ phân tán, thử nghiệm tích hợp sandwich, tích hợp đường trực, tích hợp tần số cao, tích hợp lớp, v.v.
- 

Kiểm thử tích hợp Big Bang

❖ Trong kiểm tra tích hợp Big Bang, tất cả các thành phần hoặc mô-đun được tích hợp đồng thời, sau đó mọi thứ được kiểm tra tổng thể. Theo hình ảnh trên, tất cả các mô-đun từ 'Mô-đun 1, đến' Mô-đun 6, được tích hợp đồng thời sau đó thử nghiệm được thực hiện.

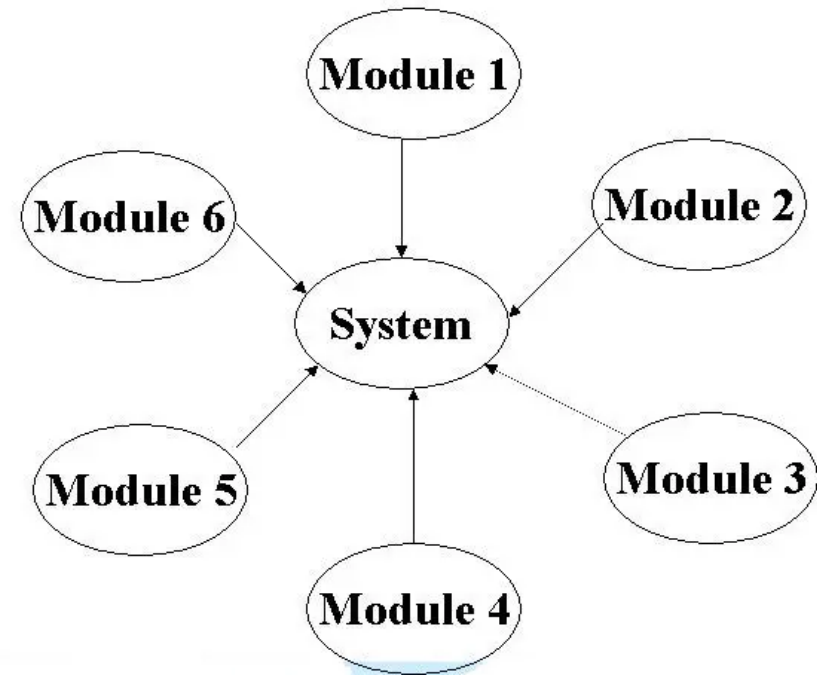
❖ Ưu điểm:

- 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.
- Mất 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.

Big Bang Integration Testing



Kiểm thử tích hợp Top-down

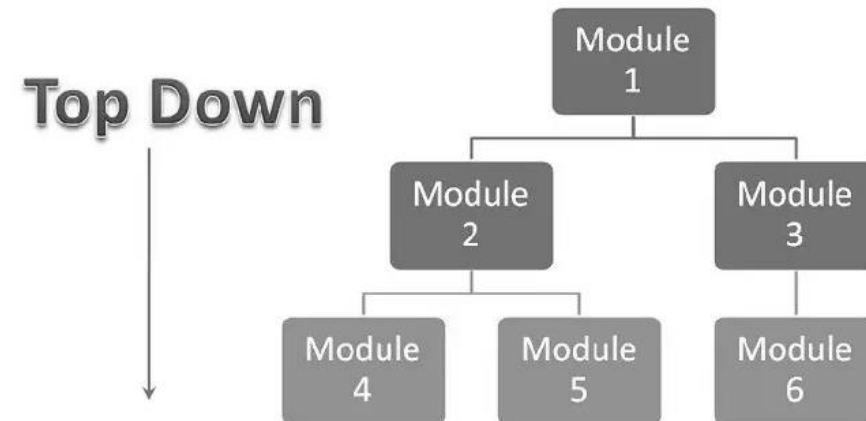
❖ Việc kiểm tra diễn ra từ trên xuống dưới, theo dòng điều khiển hoặc cấu trúc kiến trúc (ví dụ: bắt đầu từ GUI hoặc menu chính). Nó được sử dụng cho Stub testing. Dưới đây là sơ đồ của Cách tiếp cận Từ trên xuống:

❖ 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.

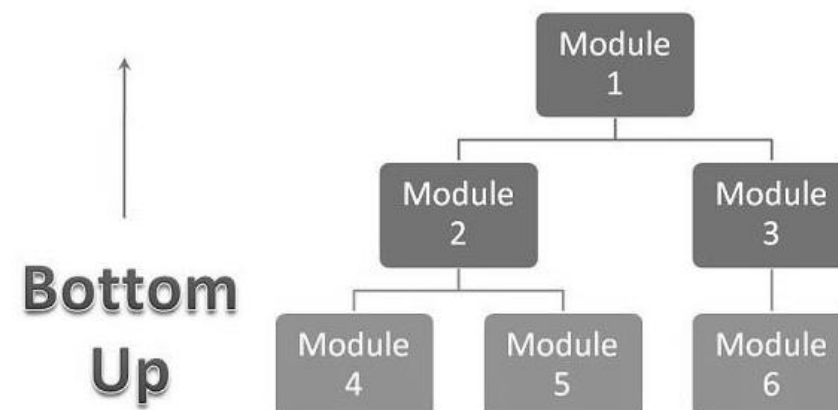
❖ Nhược điểm:

- 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.



Kiểm thử tích hợp Bottom-Up

- ❖ Mỗi module ở mức thấp hơn được thử nghiệm với các module cao hơn cho đến khi tất cả các module đều được kiểm tra. Nó được sử dụng cho Driver testing. Thể hiện bằng biểu đồ dưới đây:
- ❖ 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



Kiểm thử tích hợp gia tăng

- ❖ Trong phương pháp này, kiểm tra được thực hiện bằng cách kết hợp hai hay nhiều module có liên quan một cách hợp lý. Sau đó, các phân hệ liên quan khác được thêm vào và kiểm tra sự hoạt động đúng đắn. Quá trình tiếp tục cho đến khi tất cả các module được tham gia và thử nghiệm thành công. Quá trình này được thực hiện bằng cách sử dụng các chương trình giả gọi là Stub and Driver. Sơ khai và trình điều khiển không thực hiện toàn bộ logic lập trình các module nhưng chỉ mô phỏng giao tiếp dữ liệu với các module được gọi.
 - Stub: Được gọi bởi Module dưới Test.
 - Driver: Gọi Module để được kiểm tra.
- ❖ Ưu điểm:
 - Các khiếm khuyết được tìm thấy sớm, dễ dàng phát hiện nguyên nhân
- ❖ Nhược điểm:
 - Tốn thời gian vì Stubs và Driver phải được phát triển và sử dụng trong thử nghiệm.

Kiểm thử tích hợp Sandwich

- ❖ Kiểm thử tích hợp Sandwich kết hợp của cả hai cách tiếp cận từ trên xuống và từ dưới lên. Nó cũng được gọi là kiểm thử tích hợp lai hoặc kiểm thử tích hợp hỗn hợp. Trong Kiểm thử tích hợp Sandwich, hệ thống được tạo thành từ ba lớp:
 - Một lớp ở giữa sẽ là mục tiêu của thử nghiệm
 - Một lớp bên trên lớp đích và một lớp bên dưới lớp đích Thử nghiệm bắt đầu từ lớp ngoài và hội tụ ở lớp giữa
- Ưu điểm:
 - Các lớp trên cùng và dưới cùng có thể được kiểm tra song song.
- Nhược điểm:
 - Việc kiểm tra mở rộng các hệ thống con không được thực hiện trước khi tích hợp.

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

Kết luận

- ❖ Intergration test là 1 bước rất quan trọng trong suốt quá trình kiểm thử.
- ❖ Liệu phần mềm có được đảm bảo chất lượng hay không?
- ❖ Liệu hệ thống có vận hành theo đúng mong muốn người dùng hay không sẽ được kiểm tra qua bước này.