**1/. Trình bày các mức kiểm thử theo hiểu biết của bạn**

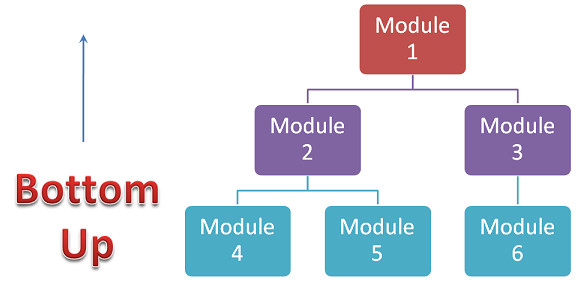
* Có **04 mức độ kiểm thử phần mềm** chính cần hoàn thành trước khi đưa phần mềm vào sử dụng: unit testing, integration testing, system testing, và acceptance testing.

## **Unit Testing (Kiểm thử đơn vị):**

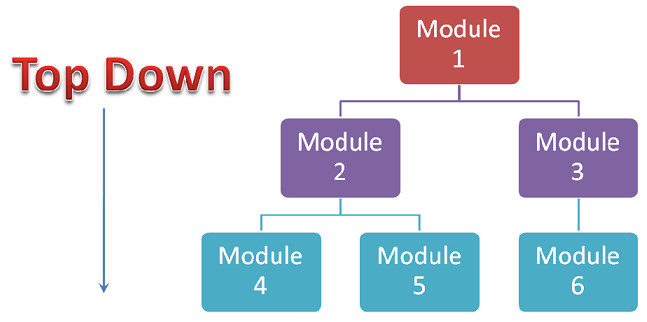
* Unit testing, còn được gọi là Component testing, mức độ kiểm thử phần mềm này thường do Developer phụ trách, họ sẽ đi kiểm tra các module, các hàm, các phương thức, các lớp,… mà họ viết ra nhằm gia tăng sự tin cậy cho các chức năng mà mình viết.
* Kiểm thử đơn vị nằm trong phạm vi của Kiểm thử hộp trắng (White-box testing), tức là kiểm tra code bên trong của một chức năng hoặc hệ thống để xem chức năng hoặc hệ thống đó được viết đúng chuẩn code hay chưa, đoạn code đó khi chạy hiệu năng có tốt hay không, có nhanh hay không, có tốn tài nguyên hay không,..
* Tuy nhiên, kiểm thử theo phương pháp này không thể tìm được mọi lỗi của ứng dụng. Không thể đánh giá mọi tình huống có thể xảy ra trong chương trình. Nó bị giới hạn bởi các tình huống và bộ dữ liệu test mà developer sử dụng để kiểm thử chương trình.

## Integration Testing (Kiểm thử tích hợp)

* Kiểm thử tích hợp là kiểm thử sự tương tác giữa các chức năng với nhau trong hệ thống và được thực hiện bởi Tester. Ví dụ: sau khi đã unit test chức năng đăng nhập và chức năng đăng ký thì ta có thể tiến hành kiểm thử tích hợp của 2 chức năng này để xem chúng có tương tác tốt với nhau không, sau khi đăng ký thành công thì ta có thể tiến hành đăng nhập bằng tài khoản đã đăng ký xem có thực hiện được không.
* Integration Testing có hai cách tiếp cận phổ biến: Tích hợp từ dưới đi lên (Bottom-up integration) và tích hợp đi trên đi xuống (Top-down integration)
  + **Tích hợp từ dưới đi lên**: Việc kiểm thử được bắt đầu bằng unit test, sau đó sẽ đi đến mức cao hơn, đó là kết hợp các thành phần và chức năng của hệ thống.

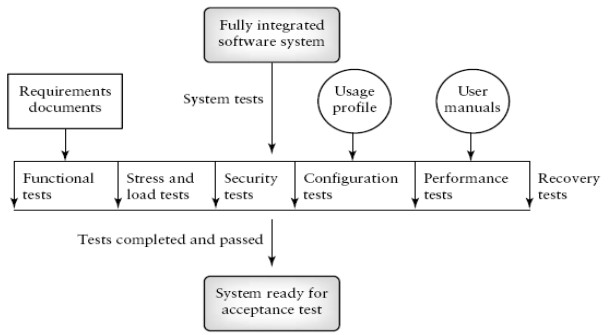


* + **Tích hợp từ trên đi xuống**: Những module cấp cao nhất được thử nghiệm đầu tiên rồi đến các module thấp hơn được thử nghiệm.



## System Testing (Kiểm thử hệ thống)

* Kiểm thử hệ thống là kiểm thử một hệ thống đã hoàn thành, đã tích hợp đầy đủ các chức năng nhằm kiểm tra xem hệ thống phần mềm đó có đáp ứng đầy đủ các yêu cầu chức năng theo bản đặc tả yêu cầu phần mềm (SRS) hay không. Người thực hiện test level này thường là Tester.
* Kiểm thử hệ thống thuộc phạm vi Kiểm thử hộp đen (tức là Tester chỉ quan tâm đầu vào và kết quả mong đợi ở đầu ra mà không cần kiểm tra code bên trong được viết như thế nào).



* Một số loại kiểm thử thường được thực hiện trong System Test:
  + - Kiểm thử chức năng (**Functional Test**): Là kiểm thử toàn bộ hệ thống, đảm bảo hệ thống hoạt động đúng theo yêu cầu được đưa ra trước đó
    - Kiểm thử hiệu năng (**Performance Test**): Là kiểm tra sự tuân thủ của hệ thống với các yêu cầu được chỉ định về hiệu năng. Xác định những thuộc tính chất lượng của hệ thống như khả năng mở rộng, độ tin cậy…
    - Kiểm thử cơ sở dữ liệu ( **Database Test**): Là kiểm tra dữ liệu hiển thị trên hệ thống có giống với dữ liệu trong cơ sở dữ liệu hay không?
    - Kiểm thử khả năng bảo mật ( **Security Test**): Là kiểm tra hệ thống được bảo vệ an toàn, không bị đánh cắp dữ liệu, thông tin trước các tấn công từ bên ngoài.
    - Kiểm thử tính khả dụng (**Usability Test**): Kiểm tra tính thân thiện với người dùng và tính dễ sử dụng của hệ thống.
    - Kiểm tra tính tương thích ( **Compatibility Test**) : Là kiểm tra xem hệ thống có tương thích với các yếu tố khác của hệ thống mà nó sẽ hoạt động hay không? (Ví dụ: Trình duyệt, hệ điều hành, phần cứng).
    - Kiểm tra khả năng phục hồi ( **Recovery Test**): Là kiểm tra hệ thống có khả năng khôi phục trạng thái ổn định khi gặp các sự cố bất thường không.

## Acceptance Testing (Kiểm thử chấp nhận)

* Acceptance Test (Kiểm thử chấp nhận) – kiểm tra xem hệ thống có đáp ứng đúng nhu cầu và mong đợi của khách hàng hay không.
* Kiểm thử chấp nhận thường là trách nhiệm của người dùng hoặc khách hàng. Trong kiểm thử hệ thống, khách hàng sẽ kiểm tra xem phần mềm được viết có hoạt động đúng như mong đợi của mình không, có đảm bảo tính tiện dụng, hiệu suất hoạt động có như mong đợi không, có bảo mật tốt hay không,….
* Tìm lỗi không phải là trọng tâm chính trong kiểm thử chấp nhận, vì việc tìm lỗi đã được đội Developer và Tester thực hiện trong các giai đoạn kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thống rồi.
* Acceptance test gồm 2 loại kiểm thử là:
  + **Alpha Test**: người dùng kiểm thử phần mềm ngay tại nơi phát triển phần mềm, lập trình viên sẽ ghi nhận các lỗi hoặc phản hồi, và lên kế hoạch sửa chữa.
  + **Beta Test**: phần mềm sẽ được gửi tới cho người dùng để kiểm thử ngay trong môi trường thực, lỗi hoặc phản hồi cũng sẽ gửi ngược lại cho lập trình viên để sửa chữa.

**2/. Trình bày các loại kiểm thử**

* Manual Testing
  + Manual testing là một loại kiểm thử phần mềm trong đó người kiểm tra thực hiện chạy test case một cách thủ công mà không dùng bất cứ một công cụ tự động nào. Manual Testing là kiểu test nguyên thủy nhất trong các loại kiểm tra giúp tìm ra lỗi trong hệ thống phần mềm. Bất kì ứng dụng nào cũng đều phải được kiểm tra một cách thủ công trước khi có thể thực hiện test tự động. Manual Testing đòi hỏi nhiều effort hơn nhưng nó là cần thiết để kiểm tra tính khả thi của tự động hóa.
  + Manual Testing không đòi hỏi kiến thức về các công cụ kiểm thử. Và một trong những vấn đề cơ bản của kiểm thử phần mềm là "không thể tự động hóa 100%" Điều này làm cho Manual Testing trở nên bắt buộc.
  + Các loại Manual Testing
    - Black Box Testing
    - White Box Testing
    - Unit Testing
    - System Testing
    - Integration Testing
    - Acceptance Testing
* Automation Testing
  + 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
    - Quick Test Profressional - (HP)
    - Selenium
    - Test Architect - (LogiGear)
    - Ranorex
    - Visual Studio CodedUI Testing
    - TestComplete (SmartBear)
    - SOAPUI - Web Services Testing (SmartBear)