**Sự khác nhau giữa Manual Testing và Automation Testing**

**1. Manual Testing là gì?**

Manual testing là việc thử nghiệm một phầm mềm hoàn toàn được làm bằng tay bởi người tester. Nó được thực hiện nhằm phát hiện lỗi trong phầm mềm đang được phát triển. Trong manual testing, tester sẽ thực hiện các trường hợp kiểm thử và tạo báo cáo kiểm thử hoàn toàn thủ công mà không có bất kỳ sự trợ giúp của công cụ tự động nào.

**2. Automation Testing là gì?**

Automation testing là phương pháp kiểm thử tự động. Người tester sẽ phải viết các kịch bản kiểm thử sau đó sử dụng các tool hỗ trợ để thực hiện kiểm thử, phương pháp này sẽ giúp việc kiểm thử hiệu quả và tốn ít thời gian hơn. Automation testing giúp chạy các kịch bản kiểm thử lặp lại nhiều lần và các task kiểm thử khác khó thực hiện bằng tay như performance testing và stress testing.

**3. Điểm khác nhau giữa Manual Testing và Automation Testing**

| **Parameter** | **Automation Testing** | **Manual Testing** |
| --- | --- | --- |
| Definition | Automation testing sử dụng các tool để thực hiện các trường hợp kiểm thử | Thực hiện kiểm thử hoàn toàn thủ công không có sự trợ giúp của bất kỳ công cụ tự động nào, được thực hiện bời tester |
| Processing time | Thời gian kiểm thử rút ngắn hơn so với manual testing | Manual testing tốn nhiều thời gian và nguồn nhân lực |
| Exploratory Testing | Không cho phép kiểm thử khám phá | Có thể kiểm thử khám phá trong manual testing |
| Reliability | Kết quả kiểm thử đáng tin cậy vì nó được thực hiện bằng các tool và các kịch bản | Kết quả kiểm thử không đáng tin cậy vì có khả năng xảy ra lỗi của con người |
| UI change | Chỉ là thay đổi nhỏ trong giao diện AUT nhưng các kịch bản kiểm thử tự động cần phải sửa đổi để hoạt động đúng như mong đợi | Những thay đổi nhỏ thư thay đổi về id, class sẽ không cản trở quá trình kiểm thử |
| Investment | Cần phải đầu tư cho các công cụ kiểm thử | Cần đầu tư về nguồn nhân lực |
| Test Report Visibility | Tất cả các bên liên quan có thể đăng nhập vào hệ thống xem kết quả đã kiểm thử | Kết quả được lưu lại trong excel hoặc word |
| Performance Testing | Được thực hiện trong kiểm thử Load testing, stress testing | Không khả thi trong kiểm thử Load testing, stress testing |
| Parallel Execution | Có thể thực hiện song song trên cấc nền tảng vận hành khác nhau và giảm thời gian thực hiện kiểm thử | Kiểm thử song song trên các nền tảng khác nhau sẽ phải tăng nguồn nhân lực |
| Programming knowledge | Yêu cầu phải có kiến thức lập trình | Không cần có kiến thức lập trình vẫn có thể thực hiện |
| Ideal approach | Automation testing rất hữu ích khi thường xuyên thực hiện chạy lại một kịch bản nhiều nhiều lần | Manual testing hữu ích khi chạy bộ test case một hoặc hai lần |

**4. Ưu, nhược điểm Manual Testing**

**Ưu điểm**

* Dễ dàng cho việc test giao diện, người tester sẽ có phản hồi nhanh và trực quan về giao diện ứng dụng
* Mất ít chi phí cho các tool tự động và quy trình
* Khi có thay đổi nhỏ manual testing manual testing không bị mất nhiều thời gian để thay đổi các trường hợp kiểm thử

**Nhược điểm**

* Kết quả kiểm thử ít tin cậy hơn vì có thể sai xót do yếu tố con người
* Qúa trình thực hiện các ca kiểm thử không được ghi lại, do vậy nó không có tính tái sử dụng
* Với một số task khó thực hiện thủ công như performance testing và stress testing thì manual testing rất khó để thực hiện

**5. Ưu, nhược điểm Automation Testing**

**Ưu điểm**

* Sử dụng tool tự động giúp tìm kiếm được nhiều lỗi hơn
* Automation testing nhanh và hiệu quả
* Qúa trình kiểm thử được ghi lại, điều đó giúp chạy lại kịch bản kiểm thử nhiều lần và thực hiện trên nhiều nền tảng khác nhau
* Automation testing được thực hiện bằng các công cụ phầm mềm, do đó nó hoạt động không mệt mỏi không giống như người kiểm thử tester
* Automation testing năng suất và chính xác
* Phạm vi kiểm thử rộng vì kiểm tra tự động không quên kiểm tra ngay cả đơn vị nhỏ nhất

**Nhược điểm**

* Rất khó có cái nhìn đúng và trực quan về giao diện người dùng như màu sắc, font chữ, vị trí, kích thước các button nếu như không có yếu tố con người
* Chi phí cho các tool kiểm thử có thể tốn kém, có thể làm tăng chi phí trong khâu kiểm thử của dự án
* Nếu có một thay đổi nhỏ cũng sẽ mất thời gian để update kịch bản kiểm thử

**Kết luận**

* Manual testing là kiểm thử phần mềm trong đó các kiểm thử được thực hiện bởi QA
* Automation testing người tester sẽ viết các kịch bản test bằng các tập lệnh và nhờ vào sự giúp đỡ của các tool tự động để thực hiện kiểm thử
* Manual testing sẽ giúp có cái nhìn trực quan và chính xác về ứng dụng
* Automation testing giúp tìm thấy nhiều lỗi hơn
* Manual testing cho kết quả ít tin cậy hơn vì con người có thể mắc lỗi
* Automation testing tiêu tốn chi phí cho các tool kiểm thử, làm tăng chi phí của dự án
* Manual testing tiêu tốn thời gian và nguồn nhân lực
* Automation testing nhanh hơn và hiệu quả hơn so với manual testing

2 các loại test

* 1) Test cấp đơn vị (Unit testing)  
  2) Test cấu hình (Shakeout testing)  
  3) Test sơ lượt (Smoke testing (Ad-hoc testing))  
  4) Test chức năng (Functional testing)  
  5) Test tích hợp (Integration testing)  
  6) Test hồi quy (Regression testing)  
  7) Test hệ thống (System testing)  
  8) Test tải dữ liệu (Load testing)  
  9) Test tải trọng (Stress testing)  
  10) Test hiệu suất (Performance testing)  
  11) Test chấp nhận từ người sử dụng (User acceptance testing)  
  12) Test hộp đen (Black box testing)  
  **13) Test hộp trắng (White box testing)**  
  14) Test Alpha (Alpha testing)  
  15) Test Beta (Beta testing)  
  (Ghi chú: Ngoại trừ kiểu test Shakeout và Unit test được thực hiện bởi nhóm quản lý cấu hình (CMT-Configuration Management Team) và người lập trình (coder/developer), tất cả các kiểu test khác được thực hiện bởi Tester QA.
* 1). Test Unit là gì? Là kiểu test kiểm tra code xem liệu chức năng nó đang thực hiện có đúng cách hay không theo như yêu cầu.
* 2). Test Shakeout là gì? Kiểu test này cơ bản là kiểu test về khả năng của hệ thống mạng, kết nối dữ liệu và sự tương tác của các module. Thông thường thì kiểu test này là do nhóm quản lý cấu hình chuẩn bị thiết lập các môi trường test thực sự. Họ cũng test xem liệu các thành phần chính của phần mềm có hoạt động bất thường không. Kiểu test này thực hiện trước khi tiến hành thực hiện trong môi trường test. Sau khi test shakeout, bước kế tiếp là test smoke (kiểu test được thực hiện bởi tester sau khi biên dịch, được tiến hành trong môi trường test).
* 3). Test smoke là gì? Là kiểu test được thực hiện khi phần code được biên dịch mới chỉ được chuẩn bị tiến hành trong môi trường test. Kiểu này cơ bản giống như kiểu ad hoc để kiểm tra đại khái để chắc rằng các chức năng chính có bị bất thường không? Nó mở đầu cho quá trình test bởi Tester QA. Sau khi test smoke, các tester sẽ thực hiện test khả năng thực hiện của các chức năng.
* 4). **Test Chức năng là gì? Là kiểu test liệu mỗi và mọi chức năng của ứng dụng đó đang làm việc có như yêu cầu của tài liệu. Nó là kiểu test chính mà 80% công việc test được thực hiện. Trong kiểu test này thì các testcase được thực hiện (hoặc thi hành).**
* 5). Test Tích hợp là gì? là kiểu test kiểm tra liệu tất cả các module là được kết hợp hoặc chưa kết hợp lại cùng với nhau thực hiện công việc có đạt được kết quả như tài liệu yêu cầu đã được xác định (do mỗi lập trình viên thực hiện trên các module khác nhau. Khi họ hoàn thành đoạn code của họ, nhóm quản lý cấu hình ráp chúng lại với nhau và chuẩn bị biên dịch. Các tester cần chắc rằng các module này bây giờ đã được kết hợp và làm việc theo như yêu cầu – tức là phải test theo như yêu cầu).
* 6). Test hồi quy là gì? Khi một chức năng mới được thêm vào phần mềm, chúng ta cần chắc chắn rằng phần chức năng mới được thêm vào không phá hỏng các phần khác của ứng dụng. Hoặc khi lỗi đã được chỉnh sửa, chúng ta cần chắc chắn rằng lỗi chỉnh sửa không phá hỏng các phần khác trong ứng dụng. Để test điều này chúng ta thực hiện kiểu test lặp đi lặp lại gọi là test hồi quy.
* 7). Test hệ thống là gì? Khi tester hoàn thành công việc test (các tester test ứng dụng trong các môi trường test, nghĩa là họ test với dữ liệu test, không test trên dữ liệu thật), ứng dụng (phần mềm) phải được test trên môi trường thật. Nó nghĩa là gì, tức là kể từ khi các tester test nó trong môi trường test với dữ liệu test, chúng ta phải chắc chắn rằng ứng dụng làm việc tốt trong môi trường thật với dữ liệu thật. Trong môi trường test, một vài điều không thể test hoặc thao tác giả. Tất cả sẽ khác nhau và cơ sở dữ liệu khác nhau, một số thao tác có thể không làm việc như mong đợi khi ứng dụng được chuyển từ môi trường test sang môi trường sản phẩm (test enviroment to production environment).
* 8). Test tải dữ liệu? Là kiểu test kiểm tra thời gian đáp lại người dùng với ứng số lượng người dùng bất kỳ trong một ngữ cảnh nào đó của cùng một ứng dụng tại cùng một thời điểm.
* 9). Test tải trọng là gì? Là kiểu test kiểm tra thời gian đáp lại người dùng với ứng số lượng người dùng bất kỳ trong nhiều ngữ cảnh khác nhau của cùng một ứng dụng tại cùng một thời điểm.