**Test case là gì?**

**Test case** hay **kịch bản kiểm thử** được dùng để kiểm tra các tình huống có thể xảy ra, từ đó xác định xem ứng dụng phần mềm hay một chức năng ứng dụng có hoạt động đúng hay không.

Sau đây là những thành phần chính bạn có thể bắt gặp khi xây dựng test case:

* Mã test case (ID test case): Giá trị cần để xác định thứ tự của test case. ID có thể bao gồm chữ và số được đánh dấu theo thứ tự tăng dần.
* Mục đích kiểm thử (Test case Description): Mô tả mục đích của test case là kiểm tra chức năng nào. Ở mục này, Tester sẽ mô tả công việc thực hiện.
* Dữ liệu kiểm thử (Test Data): Dữ liệu cần chuẩn bị để thực hiện việc kiểm thử, có thể có hoặc không tùy từng quy mô dự án. Tester có thể để ở dạng tên data hoặc đường dẫn tới file.
* Các bước thực hiện (Test Steps): Mô tả chi tiết những bước thực hiện test. Tuy nhiên, Tester nên mô tả một cách ngắn gọn và thật rõ ràng. Đồng thời không nên bỏ qua các sự kiện thiết yếu để có thể dễ dàng thực hiện lại khi có lỗi.
* Kết quả mong muốn (Expected Results): Hiển thị kết quả mong đợi từ những bước kiểm thử. Kết quả mong muốn thường dựa trên yêu cầu của khách hàng hoặc đánh giá theo tài liệu chuyên môn.
* Kết quả thực tế (Test Results): Hiển thị kết quả thực tế từ những bước thực hiện trên môi trường của hệ thống, thường sẽ là pass, fail hoặc pending.

**Một số template test case phổ biến :**

1. \*\*Test Case về Input Rỗng (Empty Input)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm xử lý như thế nào khi không có dữ liệu đầu vào.

- Bước thực hiện: Để trống các trường đầu vào hoặc gửi dữ liệu trống.

- Kỳ vọng: Phần mềm cần xử lý tình huống này một cách đúng đắn, có thể trả về thông báo lỗi hoặc yêu cầu nhập lại.

2. \*\*Test Case về Input Hợp Lệ (Valid Input)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm xử lý dữ liệu hợp lệ như thế nào.

- Bước thực hiện: Cung cấp dữ liệu đầu vào tuân thủ các quy định và ràng buộc của ứng dụng.

- Kỳ vọng: Phần mềm cần xử lý dữ liệu một cách đúng đắn và trả về kết quả mong đợi.

3. \*\*Test Case về Input Không Hợp Lệ (Invalid Input)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm xử lý dữ liệu không hợp lệ như thế nào.

- Bước thực hiện: Cung cấp dữ liệu đầu vào không tuân thủ các quy định và ràng buộc của ứng dụng.

- Kỳ vọng: Phần mềm cần xử lý tình huống này một cách đúng đắn, có thể trả về thông báo lỗi hoặc yêu cầu nhập lại.

4. \*\*Test Case về Giới Hạn (Boundary Test)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm có xử lý các trường hợp ở ranh giới tối đa và tối thiểu của các tham số không.

- Bước thực hiện: Sử dụng các giá trị tối đa và tối thiểu của các tham số.

- Kỳ vọng: Phần mềm cần xử lý các giới hạn này một cách đúng đắn và không gây lỗi.

5. \*\*Test Case về Xử Lý Ngoại Lệ (Exception Handling)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm có xử lý các tình huống ngoại lệ như lỗi kỹ thuật, sự cố mạng, v.v. như thế nào.

- Bước thực hiện: Tạo ra các tình huống ngoại lệ như mất kết nối mạng, lỗi máy chủ, v.v.

- Kỳ vọng: Phần mềm cần xử lý các tình huống ngoại lệ một cách đúng đắn và cung cấp thông báo hoặc hướng dẫn thích hợp.

6. \*\*Test Case về Hiệu Suất (Performance Test)\*\*:

- Mục tiêu: Kiểm tra xem phần mềm có hoạt động một cách ổn định và đáng tin cậy trong điều kiện tải nặng không.

- Bước thực hiện: Tạo ra tải nặng bằng cách thực hiện các thao tác hoặc truy vấn lặp đi lặp lại.

- Kỳ vọng: Phần mềm cần duy trì hiệu suất ổn định và không gây ra các vấn đề về tài nguyên hoặc thời gian đáp ứng quá lớn.

Dưới đây là một số template test case khác mà bạn có thể thêm vào danh sách kiểm tra của mình:

7. \*\*Test Case về Tích Hợp (Integration Test)\*\*:

- Mục tiêu: Kiểm tra xem các thành phần khác nhau của phần mềm có tương tác với nhau như thế nào.

- Bước thực hiện: Kết hợp các thành phần và kiểm tra kết quả khi chúng tương tác với nhau.

- Kỳ vọng: Phần mềm cần hoạt động một cách chính xác khi tích hợp các thành phần.

8. \*\*Test Case về Giao Diện Người Dùng (User Interface Test)\*\*:

- Mục tiêu: Kiểm tra tính đúng đắn và dễ sử dụng của giao diện người dùng.

- Bước thực hiện: Kiểm tra các yếu tố như bố cục, màu sắc, kích thước phông chữ, v.v.

- Kỳ vọng: Giao diện người dùng cần phải hợp lý và dễ sử dụng cho người dùng cuối.

9. \*\*Test Case về Bảo Mật (Security Test)\*\*:

- Mục tiêu: Kiểm tra các tính năng bảo mật của ứng dụng.

- Bước thực hiện: Kiểm tra các cơ chế bảo mật, kiểm tra phân quyền và xác thực, v.v.

- Kỳ vọng: Ứng dụng cần đảm bảo an toàn và không bị dễ dàng tấn công từ bên ngoài.

10. \*\*Test Case về Hiệu Năng (Load Test)\*\*:

- Mục tiêu: Kiểm tra khả năng của ứng dụng khi đối mặt với tải lớn.

- Bước thực hiện: Tạo ra tải lớn để kiểm tra tài nguyên và hiệu suất hệ thống.

- Kỳ vọng: Ứng dụng cần duy trì hiệu suất ổn định và không gây ra các vấn đề về tài nguyên.

11. \*\*Test Case về Tích Cực (Positive Test)\*\*:

- Mục tiêu: Kiểm tra các tình huống mà ứng dụng nên hoạt động một cách chính xác theo thiết kế.

- Bước thực hiện: Cung cấp đầu vào hợp lệ và kiểm tra kết quả.

- Kỳ vọng: Ứng dụng nên hoạt động đúng đắn và trả về kết quả mong đợi.