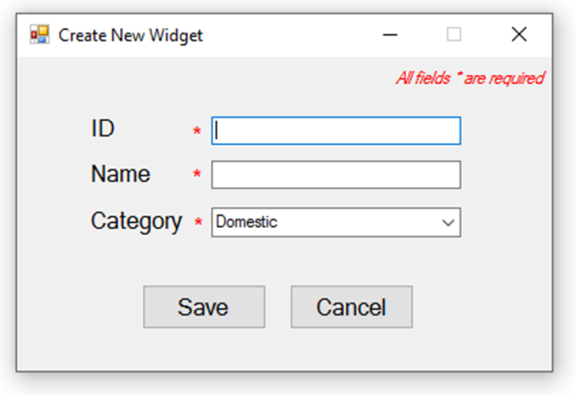
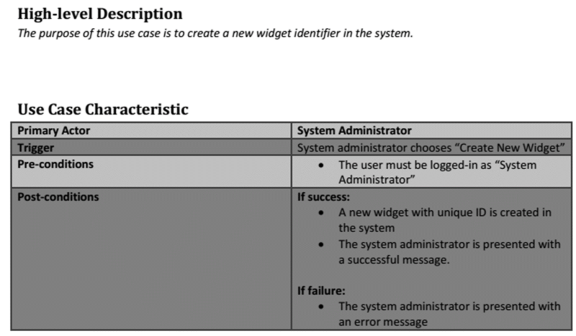
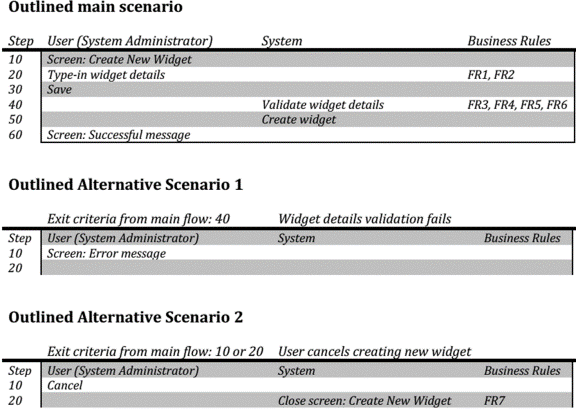
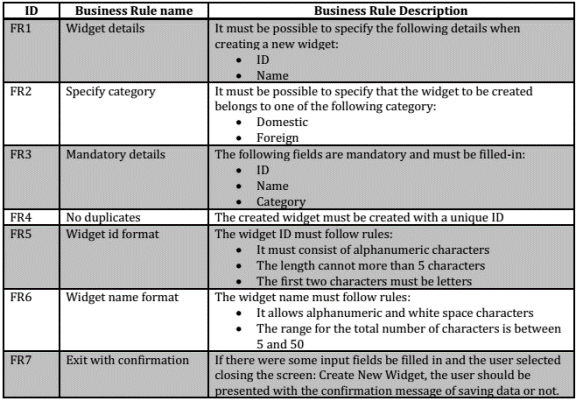
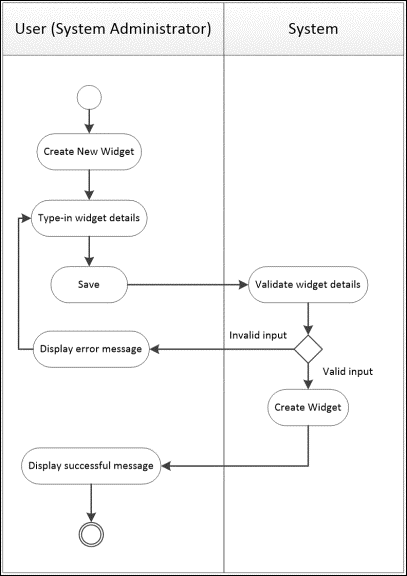
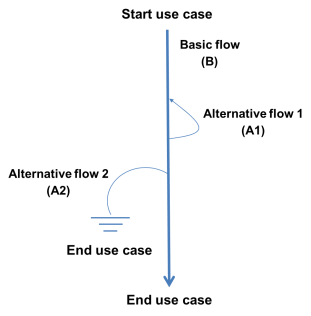
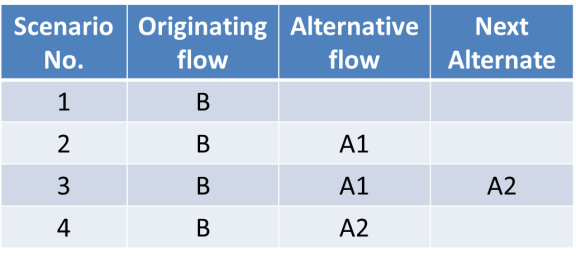
Functional Testing - Example

1. RE.
   1. Suppose we are testing a module that allows a user to enter new widget identifiers into a widget database.
      1. GUI
         1. 
      2. UC Description
         1. 
         2. 
      3. Business Rules
         1. 
2. Lưu ý
   1. Tùy thuộc vào từng dự án/tổ chức sẽ có sự thay đổi nội dung trong templates. Do đó các em có thể linh hoạt trong việc áp dụng template mà cô đề nghị.
3. Functional TCs
   1. B1. Nhận diện các kịch bản (luồng đi) có thể có trong UC
      1. Nếu chưa quen với việc đọc và phân tích tài liệu UC để nhìn ra được các kịch bản cần test, chúng ta có thể làm từng bước bằng cách
         1. 1. Vẽ ra activity diagram thể hiện luồng hoạt động của UC
            1. 
         2. 2. Thể hiện dưới dạng cây các luồng đi có thể có
            1. 
         3. 3. Đưa các luồng đi có thể có dựa trên hình vẽ ở trên vào bảng sau
            1. 
   2. B2. Thiết lập TCs cho từng kịch bản đã nhận diện ở B1
      1. Hướng dẫn
         1. Để làm được bước này chúng ta cần quay lại kiến thức đã học trong bài L06 - Black-box Testing Techniques. Tùy thuộc vào ngữ cảnh của yêu cầu mà sẽ áp dụng các kĩ thuật thiết kế TCs phù hợp. Đối với bài tập này, chúng ta sẽ áp dụng kĩ thuật ECP/BVA để thiết kế TCs.
         2. Áp dụng các kĩ thuật ECP/BVA để nhận diện TCs
            1. S1. Nhận diện input equivalence classes.

Nhận diện các điều kiện đầu vào

ĐK1. ID, Name, Category bắt buộc nhập

ĐK2. ID là duy nhất (không được trùng)

ĐK3. ID phải bao gồm cả chữ & số

ĐK4. Tổng độ dài của ID không được vượt quá 5 kí tự

ĐK5. 2 kí tự đầu của ID phải là kí tự chữ

ĐK6. Name phải bao gồm kí tự chữ và khoảng trắng.

ĐK7. Tổng độ dài của Name phải từ 5-30 kí tự

Nhận diện valid & invalid classes

ĐK1. ID, Name, Category bắt buộc nhập

Ở đây chúng ta không cần xét invalid case của field Category vì trên giao diện, field này thể hiện dưới dạng dropdown list & nó luôn có giá trị được lựa chọn. Do đó mà field này đảm bảo không bao giờ bị rỗng.

EC1.

ID, Name & Category không rỗng, valid

EC2.

ID rỗng, invalid

EC3.

ID chỉ chứa kí tự khoảng trắng, invalid

EC4.

Name rỗng, invalid

EC5.

Name chỉ chứa kí tự khoảng trắng, invalid

ĐK2. ID là duy nhất (không được trùng)

EC6.

ID không bị trùng, valid

EC7.

ID bị trùng, invalid

ĐK3. ID phải bao gồm cả chữ & số

EC8.

ID bao gồm cả chữ & số, valid

EC9.

ID chỉ chứa kí tự chữ, invalid

EC10.

ID chỉ chứa kí tự số, invalid

EC11.

ID chỉ chứa kí tự đặc biệt, invalid

ĐK4. Tổng độ dài của ID không được vượt quá 5 kí tự

EC12.

Tổng độ dài của ID <= 5, valid

EC13.

Tổng độ dài của ID > 5 kí tự, invalid

ĐK5. 2 kí tự đầu của ID phải là kí tự chữ

EC14.

2 kí tự đầu của ID là chữ, valid

EC15.

2 kí tự đầu của ID không phải là chữ, invalid

ĐK6. Name cho phép chứa các kí tự chữ, số và khoảng trắng.

EC16.

Name chứa các kí tự chữ, số và khoảng trắng, valid

EC17.

Name chứa các kí tự khác với kí tự chữ và khoảng trắng, invalid

ĐK7. Tổng độ dài của Name phải từ 5-50 kí tự

EC18.

Tổng độ dài của ID = [5-30], valid

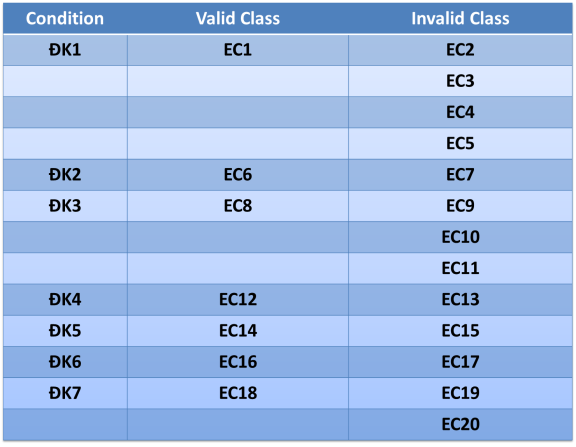
EC19.

Tổng độ dài của ID < 5 kí tự, invalid

EC20.

Tổng độ dài của ID > 30 kí tự, invalid

* + - * 1. S2&3. Đưa các classes đã nhận diện ở trên vào bảng sau



* + - * 1. S4. Thiết kế test case với bộ dữ liệu test bao trùm càng nhiều càng tốt các valid equivalence class cho đến khi tất cả các valid equivalence class đã được đưa vào trong các test cases. Các giá trị cận biên cũng được gộp vào

See also: [5 TCs có được từ S4](#a5_TCs_có_được_từ_S4)

TC1

EC1, EC6, EC8, EC12, EC15, EC17, EC19 (LB)

**LB:** **L**ower **B**oundary

Trường hợp này xét độ dài hợp lệ cận biên dưới của Name (5 kí tự)

TC2

EC1, EC6, EC8, EC12, EC15, EC17, EC19 (ALB)

**ALB:** **A**bove **L**ower **B**oundary

Trường hợp này xét độ dài hợp lệ ngay trên cận biên dưới của Name (6 kí tự)

TC3

EC1, EC6, EC8, EC12 (BB), EC15, EC17, EC19 (BUB)

**BUB:** **B**elow **U**pper **B**oundary

* Trường hợp này xét độ dài hợp lệ ngay dưới cận biên trên của Name (49 kí tự), và ngay dưới cận biên của ID (4 kí tự).

TC4

EC1, EC6, EC8, EC12 (B), EC15, EC17, EC19 (UB)

**BUB:** **B**elow **U**pper **B**oundary

Trường hợp này xét độ dài hợp lệ ngay cận biên trên của:

* ID (5 kí tự)
* Name (50 kí tự)

TC5

EC1, EC6, EC8, EC12, EC15, EC17, EC19

Trường hợp này xét độ dài hợp lệ bất kì nằm trong phạm vi cho phép của:

* ID (<= 5 kí tự)
* Name (5 - 50 kí tự)
  + - * 1. S5. Lần lượt thiết kế test case bao trùm một và chỉ một invalid equivalence class cho đến khi tất cả các invalid equivalence class đã được đưa vào trong các test cases. Các giá trị cận biên cũng được gộp vào.

See also: [Bước này liên quan đến TCs nhận diện được khi áp dụng S5 của kĩ thuật ECP/BVA ở trên](#Bước_này_liên_quan_đến_TCs_nhận_diện_được_khi_áp_dụng_S5_của_kĩ_thuật_ECP_BVA_ở_trên)

Để gộp các valid classes & invalid class nào, ta cần dựa vào bảng ở S2&3 ở trên. Xét lần lượt từng invalid class theo thứ tự từ trên xuống.

EC17, EC19 (valid) & EC2 (invalid)

Không thể gộp EC6, EC8, EC12 & EC15 (valid) với EC2 (invalid) vì tính không hợp lý về mặt nội dung điều kiện.

EC17, EC19 (valid) & EC3 (invalid)

Không thể gộp EC6, EC8, EC12 & EC15 (valid) với EC3 (invalid) vì tính không hợp lý về mặt nội dung điều kiện.

EC6, EC8, EC12, EC15 (valid) & EC4 (invalid)

Không thể gộp EC17, EC19 (valid) với EC4 (invalid) vì tính không hợp lý về mặt nội dung điều kiện.

EC6, EC8, EC12, EC15 (valid) & EC5 (invalid)

Không thể gộp EC17, EC19 (valid) với EC4 (invalid) vì tính không hợp lý về mặt nội dung điều kiện.

EC1, EC8, EC12, EC15, EC17, EC19 (valid) & EC7 (invalid)

EC1, EC6, EC12, EC15, EC17, EC19 (valid) & EC9 (invalid)

EC1, EC6, EC12, EC17, EC19 (valid) & EC10 (invalid)

EC1, EC6, EC12, EC15, EC17, EC19 (valid) & EC11 (invalid)

EC1, EC6, EC8, EC17, EC19 (valid) & EC13(BLB) (invalid)

EC1, EC6, EC8, EC15, EC17, EC19 (valid) & EC14(AUB) (invalid)

EC1, EC6, EC8, EC12, EC17, EC19 (valid) & EC16 (invalid)

EC1, EC6, EC8, EC12, EC15, EC19 (valid) & EC18 (invalid)

EC1, EC6, EC8, EC12, EC15, EC17 (valid) & EC20(BLB) (invalid)

EC1, EC6, EC8, EC12, EC15, EC17 (valid) & EC21(AUB) (invalid)

* + - 1. Đưa vào các kịch bản đã nhận diện ở B1
         1. Kịch bản 1. Basic Flow (B)

Mục đích của kịch bản này là để kiểm tra người dùng nhập dữ liệu thành công vào trong hệ thống.

5 TCs có được từ S4

* + - * 1. Kịch bản 2. B -> A1

Kịch bản này xảy ra khi người dùng nhập dữ liệu sai với những quy định (business rules) và hệ thống hiển thị thông báo cho phép người dùng nhập lại.

Bước này liên quan đến TCs nhận diện được khi áp dụng S5 của kĩ thuật ECP/BVA ở trên

* + - * 1. Kịch bản 3. B -> A1 -> A2

Kịch bản này xảy ra khi người dùng nhập dữ liệu sai với quy định (business rules) và hệ thống hiển thị thông báo cho phép người dùng nhập lại. Tuy nhiên, sau đó người dùng không nhập lại mà lại chọn đóng màn hình bằng cách thực hiện 1 trong 2 thao tác:

Nhấp nút Cancel

Nhấp vào biểu tượng 'X' ở góc trên cùng bên phải màn hình.

Đối với kịch bản này, chúng ta chỉ cần chọn lại một trong số bất kì trường hợp sai nào đã nhận diện trong kịch bản 2 để thực hiện. Và một điều cần lưu ý đến business rule FR7 về việc hiển thị thông báo yêu cầu người dùng xác nhận có muốn lưu cho dữ liệu đang nhập vào trước khi đóng form hay không.

* + - * 1. Kịch bản 4. B -> A2

Kịch bản này xảy ra khi người dùng thực hiện một trong hai thao tác sau:

Mở form nhập liệu ra nhưng không làm gì cả mà sau đó chọn đóng form theo 2 cách đã mô tả ở trên.

Nhấp nút Cancel

Nhấp vào biểu tượng 'X' ở góc trên cùng bên phải màn hình.

Mở form, sau đó nhập dữ liệu vào nhưng không chọn Save, mà chọn đóng form theo 2 cách đã mô tả ở trên. Và trường hợp này cũng lưu ý đến business rule FR7 đã đề cập trong kịch bản 3.

* + 1. Kết thúc bước này, chúng ta hoàn thiện nội dung cho sheet 'Test Requirements' của template.
       1. Mục đích
          1. Test Requirement is a statement of what should be tested in the AUT (Application Under Test). Test Requirements là những dòng mô tả về những gì nên được kiểm tra/đánh giá trong ứng dụng đang cần test.
          2. Test Requirements như là kênh trao đổi và xác nhận thông tin giữa nhóm test với bên phía khách hàng về những gì cần được kiểm tra/đánh giá cho sản phẩm/ứng dụng.
       2. Giải thích ý nghĩa của các cột trong sheet 'Test Requirements'
          1. Test Requirement ID

Mã định danh (duy nhất) các yêu cầu cần test

Tùy thuộc vào quy định đặt định danh của mỗi dự án/tổ chức.

* + - * 1. Requirement Document Reference Index

Liên hệ đến đề mục nào đã được mô tả trong tài liệu yêu cầu.

Việc đưa ra nội dung này sẽ giúp

không bị bỏ sót yêu cầu nào chưa được test

review lại tài liệu yêu cầu và giúp làm sáng tỏ những vấn đề chưa rõ ràng trong tài liệu mô tả (nếu có).

* + - * 1. Test Requirement Description

Mô tả ngắn gọn yêu cầu cần test. Nó được viết ra dựa trên các yêu cầu đặt ra trong tài liệu mô tả (use case, user story or from other sources of the software).

Trong ngữ cảnh của ví dụ này, chúng ta có thể dựa vào những kịch bản và điều kiện đầu vào đã nhận diện ở trên để hình thành các test requirements.

Với kịch bản 1 là luồng đi thành công, chúng ta xem lại các điều kiện đã mô tả trong use case làm thế nào để user có thể create new widget successfully, để từ đó hình dung ra các yêu cầu cần test.

Các trường thông tin bắt buộc nhập phải được điền đầy đủ.

Dữ liệu tạo ra phải có ID là duy nhất trong hệ thống.

Dữ liệu tạo ra phải tuân thủ theo những quy định về định dạng (format) của hai trường ID & Name.

Số kí tự nhập vào trường ID phải không được vượt quá 5 kí tự.

Số kí tự nhập vào trường Name phải không được ít hơn 5 kí tự và không được vượt quá 50 kí tự.

Người dùng không thể chọn giá trị nào khác từ trường Category ngoài 'Domestic' & 'Foreign'.

Tương tự như vậy cho các kịch bản còn lại.

Cú pháp (tiếng Anh) khi viết nội dung này

Symptom + Action + Condition

Ví dụ

User can login successfully with valid account.

* + - * 1. Test Type

Nó có thể là một trong các giá trị

Positive

Test luồng đi thành công.

Negative

Test những luồng đi có thể dẫn đến lỗi trong AUT.

Boundary

Test những trường hợp liên quan đến các điều kiện cận biên (boundary).

LooknFeel

Test những gì thuộc về giao diện người dùng

* + - * 1. Importance

Xác định độ ưu tiên của test requirements, dựa trên một số tiêu chí (Xem lại slide 22 của ST\_L07\_Functional Testing.pptx)

* 1. B3&4. Hoàn thiện mô tả TCs đã nhận diện ở B2
     1. Dựa trên những test requirements và test cases đã nhận diện ở B2, chúng ta hoàn thiện mô tả chi tiết cho từng test case bằng cách sử dụng sheet 'Test Cases' trong template.
        1. Mục đích
           1. Làm cơ sở giúp cho người tester chịu trách nhiệm thực thi (run/execute) TCs thực hiện theo những hướng dẫn chi tiết từng bước đã mô tả để kiểm tra/đánh giá các chức năng của ứng dụng đang cần test (AUT).
           2. Ngoài ra, nó cũng có thể làm cơ sở cho việc tự động hóa TCs (test automation) về sau (nếu có yêu cầu).
        2. Giải thích ý nghĩa của các cột trong sheet 'Test Cases'
           1. Test Requirement ID

Lấy lại nội dung ở sheet 'Test Requirement'

Một test requirement (TR) có thể có một hoặc nhiều test cases (TCs).

* + - * 1. Test Case ID

Mã định danh (duy nhất) cho từng test case (TC)

Tùy thuộc vào quy định đặt định danh của mỗi dự án/tổ chức.

* + - * 1. Test Objective

Mô tả ngắn gọn mục tiêu của TC này. Nó được viết ra dựa trên nội dung đã mô tả trong cột 'Test Requirement Description' của sheet 'Test Requirement'.

Cú pháp (tiếng Anh) khi viết nội dung này

Action + Function + Operating Condition

Function

may be function, feature, validation point

Condition

may be data

Action

Verify

Test

Validate

Prove

Execute

Print

Calculate

Run

... any action verb

Ví dụ

Verify that information of booked ticket are correct after booking successfully.

* + - * 1. Pre-conditions

Điều kiện tiên quyết để có thể thực thi test case này.

* + - * 1. Step Action

Mô tả từng bước tester cần thực hiện trong TC này để kiểm tra/đánh giá yêu cầu của chức năng.

* + - * 1. Test Data

Dữ liệu cần truyền vào tương ứng với từng bước (nếu có).

* + - * 1. Expected Result

Kết quả mong đợi sau khi thực hiện từng bước hoặc một số bước đã mô tả trong cột 'Step Action'.

* + - * 1. Lưu ý trên thực tế khi chạy TC, cần thêm một số thông tin khác vào template bên cạnh những thông tin đã đề cập ở trên, ví dụ:

Build

TC này được thực thi dựa vào bản build nào do bên development team chuyển sang.

Test Result

Passed or Failed

...