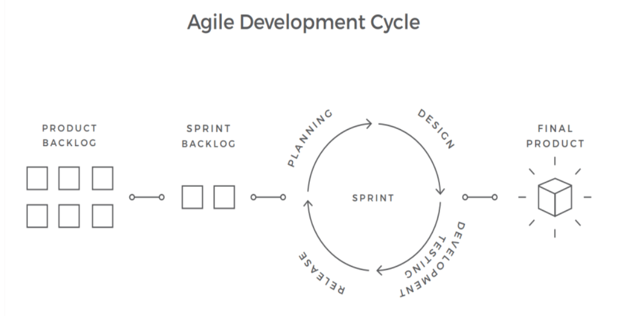
**TỔNG QUAN KIỂM THƯ PHẦM MỀM**

* **Khái niệm?**

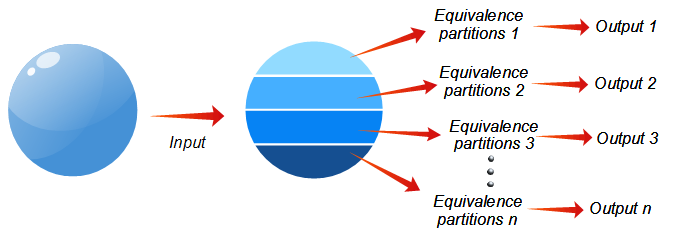
Là một quá trình thao tác thực hiện kiểm tra chương trình phần mềm với mục đích:

* Tìm lỗi phần mềm
* Chứng minh phần mềm hoạt động đúng
* **Quy trình kiểm thử phần mềm**
* Lập kế hoạch và kiểm soát (planning and control)
* Planning bao gồm các nội dung:
* Xác định phạm vi (scope), rủi ro (risk), và mục đích việc kiểm thử.
* Phương pháp kiểm thử (test approach).
* Xác định tài nguyên kiểm thử (testing resource) như: Con người, môi trường…
* Lên kế hoạch phân tích kiểm thử, thiết kế kiểm thử, thực hiện kiểm thử và đánh giá.
* Xác định điều kiện kết thúc (exit criteria).
* Control:
* So sánh tiến độ thực tế so với kế hoạch, thực hiện báo cáo tình trạng sai lệch so với kế hoạch.
* Đồng thời đưa ra các hành động cần thiết để đáp ứng được mục tiêu và nhiệm vụ của dự án.
* Phân tích và Thiết kế (analyst and design)
* Bao gồm các nội dung:
* Kiểm tra lại các thông tin kiểm thử cơ bản (test basic) bao gồm: Yêu cầu, tài liệu thiết kế, tài liệu phân tích rủi ro...
* Xác định các điều kiện kiểm thử
* Thiết kế test case
* Thiết kế môi trường kiểm thử, xác định cơ sở hạ tầng và công cụ cần thiết
* Hoàn thiện và thực hiện kiểm thử (Test implement and execution)
* Implement:
* Phát triển và đánh độ ưu tiên cho các Test case và tạo dữ liệu test (test data)
* Tạo ra bộ kiểm thử (test suites) từ các Test case
* Hoàn thiện và kiểm tra môi trường.
* Execution:
* Thực hiện bộ test suites hoặc từng Test case riêng lẻ, đảm bảo theo trình tự thời gian của dự án.
* Ghi lại kết quả của việc thực hiện test, phiên bản phần mềm sử dụng.
* So sánh kết quả thực tế (actual results) và kết quả mong đợi (expected results)
* Đánh giá điều kiện kết thúc và báo cáo (Evaluating exit criteria and reporting)
* Evaluating exit criteria: là quá trình đánh giá xem khi nào sẽ dừng việc test.
* Exit criteria sẽ được thực hiện khi:
* Một lượng lớn test case được thực hiện với tỷ lệ % pass nhất định
* Tỷ lệ lỗi ở dưới mức nhất định
* Khi đến hạn (deadlines)
* Exit criteria có nhiệm vụ:
* Đánh giá xem có cần test thêm hoặc có cần thay đổi điều kiện kết thúc.
* Dùng để viết báo cáo tóm tắt kiểm thử (test summary report) cho các bên liên quan.
* Các hoạt động kết thúc việc kiểm thử (Test closure activities)
* Test closure activities: Là các hoạt động kết thúc việc kiểm thử được thực hiện khi phần mềm sẵn sàng chuyển giao.
* Nhiệm vụ:
* Đảm bảo rằng tất cả các sản phẩm đã được plan đã được giao (deliver) và tất cả các báo cáo về sự cố đã được giải quyết.
* Hoàn thiện và lưu trữ các bộ dữ liệu test: Test script, tài liệu về môi trường test… để sử dụng sau này.
* Chuyển giao lại bộ dữ liệu test cho đội bảo trì
* Rút ra bài học cho các dự án sau này.
* **Mô hình kiểm thử sử dụng tại công ty: Mô hình Agile:**
* Phương thức phát triển phần mềm Agile là một tập hợp các phương thức phát triển lặp và tăng dần. Trong đó, các yêu cầu và giải pháp được phát triển thông qua sự liên kết cộng tác giữa các nhóm tự quản và liên chức năng.
* Agile là cách thức làm phần mềm linh hoạt để làm sao đưa sản phẩm đến tay người dùng càng nhanh càng tốt càng sớm càng tốt.
* Agile được xem như là sự cải tiến so với những mô hình cũ.



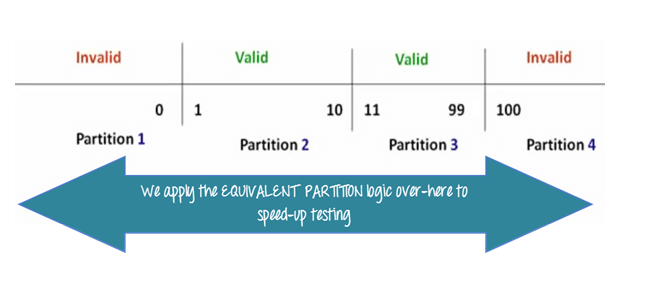
Hình 2.1: Mô hình Agile

* **Các kỹ thuật kiểm thử phầm mềm**
* *Phân vùng tương đương:*
* Định nghĩa: là phương pháp chia dữ liệu đầu vào thành các vùng tương đương nhau. Tất cả các giá trị trong một vùng tương đương sẽ cho một kết quả đầu ra giống nhau.



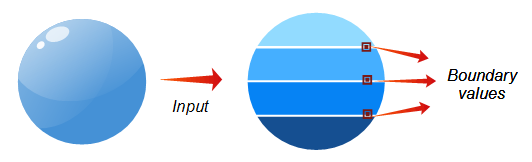
Hình 2.2: Mô hình phân vùng tương đương

* Tiến hành theo 2 bước:
* B1: Xác định các lớp tương đương theo đầu vào hoặc đầu ra. Với mỗi điều kiện đầu vào hoặc đầu ra được mô tả trong đặc tả yêu cầu thì lấy ra
* Lớp thoả mãn điều kiện
* Lớp không thoả mãn điều kiện
* B2: Thiết kế TC dựa vào các lớp tương đương trên.



Hình 2.3: Kỹ thuật phân vùng tương đương

* *Phân tích giá trị biên*



Hình 2.4: Mô hình phân tích giá trị biên

* Là tiến trình lựa chọn test case (hoặc test data) bằng cách tìm hiểu ranh giới giữa điều kiện hợp lệ và không hợp lệ.
* Check cận trên và cận dưới ngoài các điểm ranh giới
* Cách làm:
* Xác định các điểm biên [a,b]
* Thiết kế TC dựa vào các điểm biên



Hình 2.5: Kỹ thuật phân tích giá trị biên

* *Bảng quyết định*

Các bước thực hiện:

Bước 1: Liệt kê các điều kiện đầu vào

|  |
| --- |
| **Điều kiện** |
| Khách hàng mới (15%) |
| Vip card  (10%) |
| Coupon  (20%) |

Bước 2: Tính toán số lượng các kết hợp (Rules)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Điều kiện | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 |
| Khách hàng mới (15%) |  |  |  |  |  |  |  |  |
| Vip card (10%) |  |  |  |  |  |  |  |  |
| Coupon(20%) |  |  |  |  |  |  |  |  |
| Kết quả |  |  |  |  |  |  |  |  |
| Giảm giá (%) |  |  |  |  |  |  |  |  |

Bước 3: Đặt toàn bộ các kết hợp vào bảng

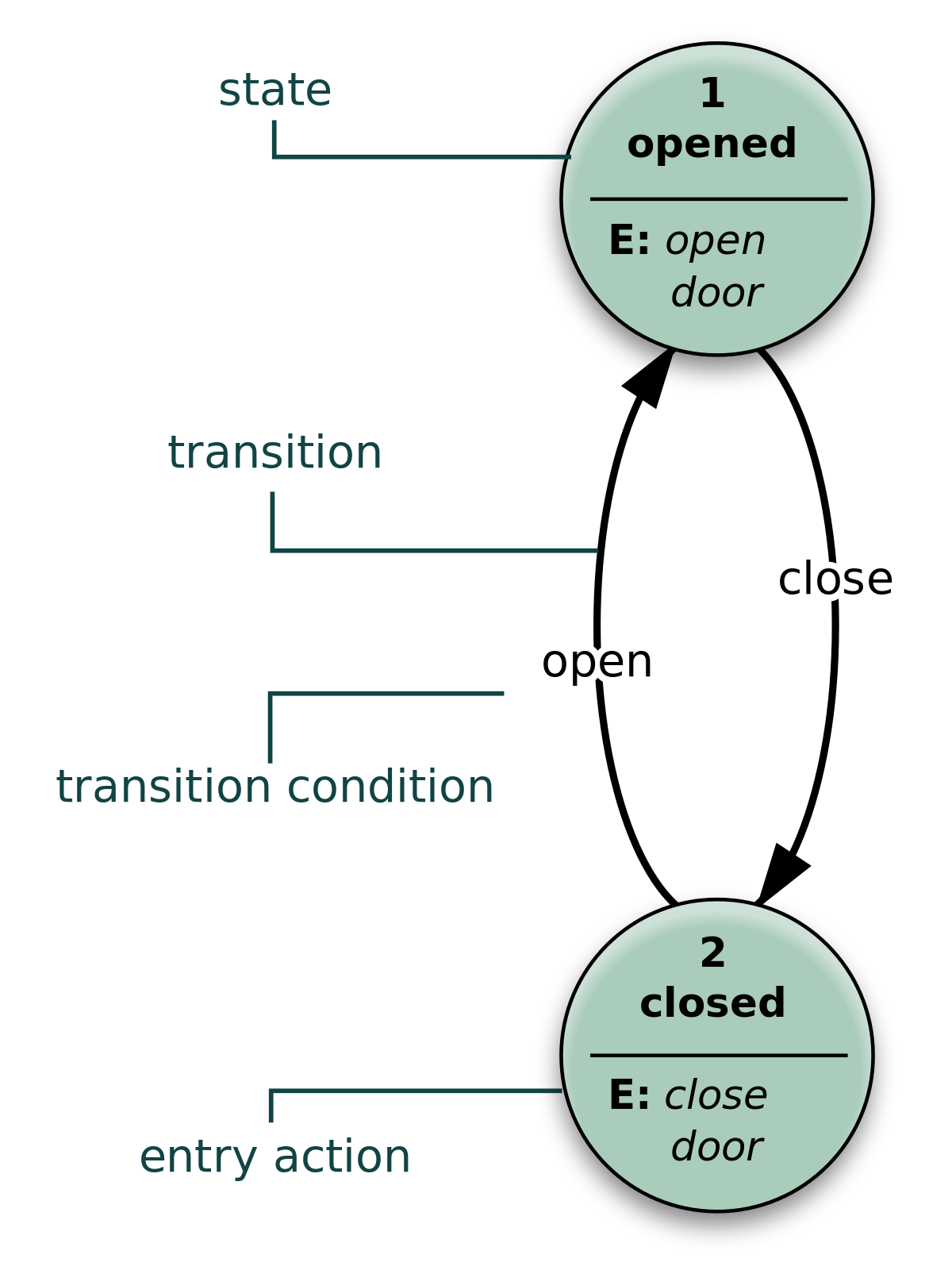
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Điều kiện | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 |
| Khách hàng mới (15%) | T | T | T | T | F | F | F | F |
| Vip card (10%) | T | T | F | F | T | T | F | F |
| Coupon(20%) | T | F | T | F | T | F | T | F |
| Kết quả |  |  |  |  |  |  |  |  |
| Giảm giá (%) | X | X | 20(?) | 15 | 30 | 10 | 20 | 0 |

Bước 4: Giảm số lượng các kết hợp và chọn Test case

* Nhìn vào Bảng ra quyết định ta có thể thấy: R1 và R2 có kết quả tương đương nhau với cùng điều kiện → Có thể giảm một trong hai TC này.
* Như vậy, nhìn vào bảng ta có thể chọn các TC như sau: R1, R3, R4, R5, R6, R7.
* *Sơ đồ chuyển trạng thái (State transition diagram)*

Mô tả:

* Sơ đồ chuyển trạng thái tập trung vào việc test sự chuyển đổi trạng thái của một đối tượng từ **trạng thái này** sang một **trạng thái khác**.
* Sơ đồ chuyển trạng thái áp dụng cho hệ thống:
* Có hữu hạn trạng thái
* Có chuỗi các sự kiện xảy ra, kết quả của trạng thái sau phụ thuộc vào kết quả của các trạng thái trước



Hình 2.6: Mô hình một sơ đồ chuyển trạng thái

Gồm 4 phần cơ bản:

* Các trạng thái (hữu hạn)
* Các sự dịch chuyển trạng thái
* Các sự kiện kích hoạt sự dịch chuyển trạng thái
* Các hành động là kết quả của của sự dịch chuyển
* **Cách log bug**
* Các thông tin của một bug report:
* The title
* Precondition
* Procedure/Steps
* Expected results
* Actual Results
* Attachment
* Cách viết title
* Title - là bộ mặt (face) của bug report. Một title tốt có thể chứa nội dung tổng quát của một bug.
* Cần tránh các tiêu đề chung chung. Ví dụ như:
* Chức năng A hoạt động không tốt/không hoạt động
* Có vấn đề với chức năng B
* Chức năng C bị có lỗi
* Hãy mô tả rõ về việc chức năng ‘not working’ như thế nào. Ví dụ:
* Thay vì viết: *Function create account does not working*
  + Hãy viết: *Display error message on clicking the SAVE button while creating a new the user, hence unable to create a new user in the application.*
* Thay vì viết: *có vấn đề với GUI*
  + Hãy viết: GUI đang bị sai font chữ
* Cách viết Procedure/Steps
* Procedure/Steps là phần thân (body) của bug report.
* Mục đích: Giúp người đọc tái hiện (reproduce) được bug
* Chú ý: Nên đánh số cho từng step để developer có thể dễ dàng thực hiện lại
* Cách viết Expected
* Expected result: Là kết quả mong đợi khi thực hiện các steps của phần Procedure
* Chú ý: Expected result nên đánh số tương ứng theo bước thực hiện
* Cách viết Actual result
* Actual result: Là kết quả thực tế khi thực hiện các steps của phần Procedure
* Chú ý: Actual result nên đánh số tương ứng theo bước thực hiện
* Attachment
* Nên capture screenshot lỗi khi thực hiện test để làm evidence cho developer