

Nhập môn công nghệ phần mềm Kiểm thử phần mềm

GV: ThS. Ngô Tiến Đức



Nội dung chính

- Chất lượng phần mềm
- Nhóm SQA
- Kiểm thử
- Kiểm thử phi thực thi
- Kiểm thử thực thi



Chất lượng phần mềm

- Phần mềm "xuất sắc"?
- Chất lượng phần mềm thể hiện ở khả năng đáp ứng yêu cầu
- Người làm phần mềm có trách nhiệm đảm bảo phần mềm chạy đúng
- -> Chất lượng phần mềm phải được đảm bảo ngay từ đầu



Nhóm SQA (1)

Thành viên nhóm SQA (đảm bảo chất lượng phần mềm):

- Đảm bảo các thành viên phát triển phần mềm đều làm việc với chất lượng cao:
 - Tại cuối mỗi workflow
 - Tới khi sản phẩm hoàn thiện
- Đảm bảo quy trình làm việc (VD: đáp ứng các chuẩn)



Nhóm SQA (2)

Quản lý nhóm SQA:

- Độc lập với nhóm phát triển phần mềm
- Không nhóm nào có quyền cấp trên



Nhóm SQA (3)

- Quản lý cấp cao của cả 2 nhóm mới có quyền quyết định:
 - Bàn giao sản phẩm đúng thời hạn nhưng có lỗi
 - Kiểm thử thêm và bàn giao trễ
- Quyết định phải dựa trên lợi ích của khách hàng và công ty



Kiểm thử (1)

"V & V":

- Verification (xác minh): Cuối workflow
 - Kiểm tra tính đúng đắn và tương thích
- Validation (xác nhận): Cuối dự án
 - Kiểm tra đúng yêu cầu



Kiểm thử (2)

2 dạng kiểm thử:

- Kiểm thử thực thi (Execution-Based Testing)
- Kiểm thử phi thực thi (Non-Execution-Based Testing)



Kiểm thử thực thi (1)

- Nguyên lý Dijkstra (1972): "Program testing can be a very effective way to show the presence of bugs, but it is hopelessly inadequate for showing their absence."
- Có những công ty dành đến 50% chi phí dự án cho kiểm thử, nhưng sản phẩm khi bàn giao vẫn có lỗi



Kiểm thử thực thi (2)

Các trường hợp kiểm thử (test case):

- Viết trước khi có code và lưu lại sau khi kiểm thử
- Cấu trúc:
 - Tập dữ liệu đầu vào
 - Kịch bản các thao tác thực hiện
 - Kết quả mong đợi



Kiểm thử thực thi (3)

Những vấn đề cần kiểm thử:

- Correctness Tính đúng đắn
- Utility Tính hữu dụng
- Reliability Tính tin cậy
- Robustness Tính ổn định
- Performance Tính hiệu quả



Kiểm thử thực thi (4)

Correctness: Phần mềm thỏa mãn các yêu cầu

- Độc lập với sử dụng tài nguyên máy tính
- Đầu vào và đầu ra giống như trong tài liệu đặc tả
- -> Kỹ thuật: Black-box testing



Kiểm thử thực thi (5)

Utility: Mức độ sản phẩm đáp ứng nhu cầu của người dùng

- Dễ sử dụng
- Chức năng cần thiết
- Giá chấp nhận được



Kiểm thử thực thi (6)

Reliability: Tần suất lỗi và độ nghiêm trọng ở mức chấp nhận được

- Thời gian trung bình giữa 2 lần lỗi liên tiếp
- Thời gian trung bình để sửa 1 lỗi
- Chi phí và thời gian để khắc phục hậu quả của lỗi



Kiểm thử thực thi (7)

Robustness: Khả năng chống lỗi

- Phạm vi môi trường để sản phẩm có thể hoạt động ổn định là rộng
- Không được có lỗi khi đầu vào hợp lệ
- Không gặp sự cố khi hoạt động ngoài phạm vi môi trường



Kiểm thử thực thi (8)

Performance: Mức độ tiêu tốn tài nguyên

- Đáp ứng các ràng buộc về bộ nhớ và thời gian
- Vấn đề tối ưu phần mềm



Kiểm thử thực thi (9)

Ai là người thực hiện kiểm thử thực thi?

- "Programming is constructive, testing is destructive"
- -> Lập trình viên không nên kiểm thử code của chính mình
- Lập trình viên kiểm thử tài liệu
- Tester kiểm thử code
- Lập trình viên sửa lỗi và kiểm tra lại



Kiểm thử phi thực thi (1)

- Nguyên tắc:
 - Không tự đánh giá sản phẩm của chính mình
 - Dựa trên sức mạnh tổng hợp của nhóm
- Hai phương pháp:
 - Walkthrough (rà soát)
 - Inspection (thẩm định)



Kiểm thử phi thực thi (2)

Walkthrough: Nhóm 4-6 người

- Đại diện của workflow hiện tại và tiếp theo
- Đại diện nhóm SQA <- phụ trách (chair)
- Các thành viên đọc trước tài liệu và chuẩn bị danh sách:
 - Các vấn đề chưa hiểu
 - Các vấn đề có vẻ không đúng



Kiểm thử phi thực thi (3)

Walkthrough: Phát hiện lỗi, không sửa lỗi!

- Kéo dài không quá 2 giờ
- Sửa lỗi ngay lúc đó không đảm bảo hiệu quả
- Không phải mọi vấn đề nêu ra đều là lỗi
- -> Tập trung vào tài liệu, không phải người tham gia
- -> Không sử dụng để đánh giá năng suất làm việc



Kiểm thử phi thực thi (4)

Inspection: 4 thành viên

- Người điều hành (moderator)
- Một đại diện workflow hiện tại
- Một đại diện workflow tiếp theo
- Một đại diện nhóm SQA



Kiểm thử phi thực thi (5)

Inspection: 5 bước

- Overview: Giới thiệu và phát tài liệu thẩm định tới các thành viên
- Preparation: Đọc hiểu tài liệu và xem xét danh sách lỗi từ các dự án trước
- Walkthrough: Rà soát lại tài liệu
- Rework: Sửa lỗi
- Follow-up: Moderator kiểm duyệt lại lần cuối



Kiểm thử phi thực thi (6)

Inspection: Danh sách lỗi

- "Checklist of potential faults"
- Thống kê lỗi:
 - Theo mức độ nghiêm trọng: "Major" hay "minor"
 - Theo kiểu lỗi



Kiểm thử phi thực thi (7)

Inspection: Thống kê lỗi

- So sánh tỉ lệ lỗi với các dự án trước
- Nếu số lỗi của một artifact nhiều bất thường -> cân nhắc làm lại từ đầu
- Bảo lưu số liệu cho workflow kế tiếp
- Không sử dụng để đánh giá năng suất làm việc: "Kill the goose that lays the golden eggs"



Kiểm thử phi thực thi (8)

- Tác dụng: Phát hiện lỗi sớm
- Tìm được nhiều lỗi hơn chứng tỏ:
 - Đội rà soát/thẩm định làm việc hiệu quả hơn?
 - Đội phát triển làm việc kém đi?



Bài tập về nhà

Trả lời câu hỏi từ 41 đến 50 trong ngân hàng câu hỏi thi



Tài liệu tham khảo

- Stephen R. Schach. *Object-Oriented and Classical Software Engineering*. 8th Edition, WCB/McGraw-Hill, 2010
- T. Đ. Quế, N. M. Hùng. *Bài giảng Nhập môn công nghệ phần mềm*. HVCNBCVT, 2020