

Chương V. LỖI PHẦN MỀM (SOFTWARE ERROR)

1. Lỗi phần mềm là gì?

- Là lỗi xảy ra **trong quá trình sử dụng PM** gây ra những **kết quả (thực tế) sai lệch so với kết quả mong đợi**.
 - Cũng có thể PM chạy được, không phát sinh lỗi nhưng không đáp ứng KQ mong đợi thì cũng coi là lỗi PM.
- **Số lượng lỗi** sẽ phản ánh chất lượng PM không tốt, nhưng **không thể đánh giá được khả năng của developer**.
- **Các loại lỗi phổ biến:**
 1. User interface - lỗi giao diện, font chữ, layout bị vỡ
 2. Error handling – xử lý lỗi – sử dụng các lệnh try catch throw, trong quá trình xử lý ngoại lệ này ta xử lý không đúng.
 3. Boundary-related – lỗi liên quan đến giá trị biên - điểm nhập vào $< 0 || > 10$, cần KT các trường hợp điểm nằm trong miền này và ngoài miền này; nhập tuổi để trở thành sinh viên...
 4. Calculation – lỗi liên quan đến tính toán – công thức ta tính sai, viết sai.
 5. Initial and later states – lỗi liên quan đến khởi tạo các giá trị/ trạng thái ban đầu.
 6. Control flow – lỗi liên quan đến luồng điều khiển – chẳng hạn không theo đúng thứ tự luồng chạy mà ta mong muốn.
 7. Handling or interpreting data – lỗi liên quan đến xử lý dữ liệu.
 8. Race conditions
 9. Load conditions – lỗi trong quá trình load các website
 10. Hardware/environment Compatibility – lỗi tương thích phần cứng, phần mềm.
 11. Source, version, ID control – lỗi về phiên bản, mã nguồn, ID.

Lỗi về version: phần mềm có nhiều version, chẳng hạn v1 có một số lỗi, v2 đã fix thành công, đến v3 các lỗi này lại xuất hiện trở lại. Nguyên nhân do trong quá trình quản lý source code không tốt, xảy ra các vấn đề nhầm lẫn trong việc lấy các version cũ (chẳng hạn thay vì lấy v2 để phát triển v3, lại lấy v1 để phát triển), dẫn đến lỗi cũ còn tồn đọng.
→ Cần sử dụng những công cụ hỗ trợ quản lý các phiên bản, ví dụ GitHub/GitLab, giúp lần quét sự thay đổi của các phiên bản.
 12. Testing – lỗi liên quan đến kiểm thử - trong quá trình kiểm thử vẫn có sai sót, chẳng hạn thiết kế TCs không đúng, thiếu, bỏ sót lỗi...
 13. Documentation – lỗi tài liệu – tài liệu HDSD không khớp với cách vận hành phần mềm.

2. Những việc cần làm trong quá trình tìm lỗi:

- **Tái tạo lỗi (reproducing software error)**
 - Tái tạo lỗi là mô phỏng lại cách sinh ra lỗi.
 - Viết tất cả mọi thứ đã làm
 - Sử dụng công cụ như capture/replay program, debugger, video tape,...
 - Giao tiếp với programmer hoặc đọc code
 - Xem lại các báo cáo tương tự

- 1 số lỗi **có khả năng tái tạo**, nhưng vẫn có 1 số **không thể/rất khó** tái tạo
 - o Lỗi liên quan đến bộ nhớ động, Cấu hình(phần cứng/phần mềm), Luồng dữ liệu/luồng điều khiển, đa tiến trình... Hoặc các lỗi xảy ra chỉ đúng 1 lần → Khó tái tạo
- Lỗi luôn xảy ra trong 1 điều kiện nhất định
- Khi tìm lỗi thường quan tâm đến **triệu chứng của lỗi**
- Giả định nguyên nhân → tái tạo lại **điều kiện gây ra lỗi**
- Khi tái tạo lỗi cần tuân thủ **nguyên tắc**: đảm bảo liệt kê chính xác lỗi, liệt kê chính xác các bước để dẫn đến lỗi.
- Nếu lỗi không tái tạo được → báo cáo toàn bộ các bước đã thực hiện, kèm theo hình ảnh, quay video... làm minh chứng.
- **Phân tích lỗi**
 - Mô tả rõ ràng làm thế nào chương trình rơi vào trạng thái xác định
 - Lỗi này có triệu chứng gì, đưa ra các giả định...
 - Kiểm tra các bước trong báo cáo để chắc chắn lỗi có thể được tái tạo
- **Báo cáo lỗi**: Viết rõ ràng, chi tiết, Đánh số duy nhất, Đơn giản, Dễ hiểu, Có khả năng tái tạo, **Chỉ nêu sự thật**, không phán đoán, không nêu cảm nhận chủ quan
 - o **Cần dựa vào tài liệu yêu cầu để kiểm thử.**
- **Nội dung báo cáo**
 - **Tóm tắt (summary – quan trọng)**: viết ngắn, <20 từ, phải mô tả đầy đủ lỗi đó
 - o Khi dev đọc vào sẽ hiểu & xác định được lỗi ngay
 - o Đảm bảo độ chính xác
 - o **Gồm 3 thành phần: Triệu chứng lỗi (lỗi như thế nào) + Hành động dẫn đến triệu chứng đó + Môi trường (hoặc Điều kiện)**
 - **Mô tả chi tiết (description – quan trọng)**: môi trường test (hệ điều hành gì, trình duyệt nào), **các bước tái tạo** (dev sẽ thực hiện lại các bước này để lỗi này có xuất hiện thực sự hay không, & họ xem đó có phải là lỗi hay không để fix), **KQ mong đợi, KQ thực tế quan sát được**
 - **Mức độ nghiêm trọng (severity)** (tùy template): có thể có 3 level, 5 level... **(Critical, Serious, Minor...)** → setting mức độ nghiêm trọng sẽ tùy thuộc vào tester → Dựa vào đây dev sẽ quyết định độ ưu tiên để sửa lỗi...
 - **Tần suất xuất hiện (frequency)**: **Always** (lúc nào cũng xuất hiện), **Often, Sometimes, Rarely** (hiếm khi xuất hiện)...
 - **Độ ưu tiên để sửa lỗi (Priority)**: **Critical, Serious, Minor...** → Đánh giá mức độ quan trọng của lỗi → cần sửa liền hay chưa hay để qua phiên bản tiếp theo?
 - **Từ khóa để phân loại bug**
 - **Date, reporter**
 - **Quyết định**

3. Mẫu bug report

Về chương trình giải phương trình bậc nhất.

BugID	Status	Component	Summary	Description	Severity	Frequency	Priority	Attachment	Date	Reporter	Build
B1.001	New	Content	The application is closed (triệu chứng) when clicking on 'làm lại' button (hành động) trên giao diện chính (môi trường: cửa sổ nào, menu nào, page nào, nhóm chức năng nào...)	Test Environment: - Platform: WinXP - Server: Steps to reproduce: (các bước tái tạo, làm như thế nào để dẫn đến lỗi) 1. Click to file 'Giai_PTB2.exe' to open program 2. Click on 'Làm lại' button Actual Result: (KQ thực tế) The application is closed Expected Result: (KQ mong đợi) + The application isn't closed + Value in textbox 'Hệ số A' as default + Value in textbox 'Hệ số B' as default + Value in textbox 'Hệ số C' as default	Serious	Always	Major	(đính kèm hình ảnh, video... liên quan đến lỗi nếu có)	17/08/2011	Trần Thị Mỹ Tiên	Build 01

BugID	Status	Component	Summary	Description	Severity	Frequency	Priority	Attachment	Date	Reporter	Build
B1.002	New	Content	No error message as "Value A, B, C must be number. Please input again" is displayed (triệu chứng) after clicking on 'Giải PT' button (hành động) with invalid data (điều kiện	<p>Test Environment:</p> <ul style="list-style-type: none"> - Platform: WinXP - Server: <p>Steps to reproduce:</p> <ol style="list-style-type: none"> 1. Click to file 'Giai_PTB2.exe' to open program 2. Input value into textbox 'Hệ số A': a 3. Input value into textbox 'Hệ số B' :b 4. Input value into textbox 'Hệ số C': c (cần đưa vào dữ liệu nhập vào nếu có, đưa đúng giá trị mà tester đã nhập, để dev biết cách fix) 5. Click on 'Giải PT' button <p>Actual Result:</p> <ul style="list-style-type: none"> + No error message as "Value A or B or C is number. Please input again" is displayed + Result in textbox 'Kết quả': 'Phương trình có nghiệm NaN' <p>Expected Result:</p> <ul style="list-style-type: none"> + Error message as "Value A or B or C is number. Please input again" is displayed + Result in textbox 'Kết quả': empty 	Critical	Always	Critical		19/08/2011	Trần Thị Mỹ Tiên	Build 01