

# Bug report template

Công cụ kiểm chứng phần mềm

Hồ Tuấn Thanh

[htthanh@fit.hcmus.edu.vn](mailto:htthanh@fit.hcmus.edu.vn)

# Nội dung

- ☐ Bug report
- ☐ Test summary report
- ☐ Vòng đời lỗi phần mềm



# Test case execution status

- ❑ PASSED: actual result đúng với expected result
- ❑ FAILED: actual result ko đúng với expected result
- ❑ BLOCKED: chưa thể thực thi test case vì precondition chưa thỏa
- ❑ SKIPPED: ko tương thích, chưa có sẵn, ngoài phạm vi kiểm thử
  - VD: yêu cầu release này chạy 1000 test case, trong khi đang có 1500 test case thì 500 test case là SKIPPED
- ❑ NOT YET TESTED: chưa thực thi

# Bug/Defect report

- ❑ Với mỗi test case FAILED, sẽ có báo cáo lỗi tương ứng
- ❑ Cung cấp thông tin chi tiết về sự cố hoặc lỗi cho những bên liên quan
  - Người phát triển (developer): sửa lỗi
  - Người quản lí (PM, leader): quyết định tài nguyên, cấp phát, ưu tiên
  - Nhân viên hỗ trợ kĩ thuật: nắm bắt thông tin thực hiện, chuẩn bị
  - Kiểm thử viên (tester): biết được trạng thái của hệ thống

# Nội dung bug report

1. Bug ID
2. Function name
3. Problem summary
4. How to reproduce it
5. Reported by
6. Date
7. Assign to
8. Status
9. Priority
10. Severity

# 1. Bug ID

- ❑ Mã định danh, phân biệt các bug khác
- ❑ Bug ID khác với test case ID

## 2. Function name

- ❑ Tên chức năng có lỗi được phát hiện
- ❑ Ví dụ:
  - Đăng nhập
  - Đăng kí tài khoản
  - Xem danh sách tài khoản
  - Thêm tài khoản
  - Xóa tài khoản



# 3. Problem summary

- ❑ Tóm tắt nội dung lỗi
- ❑ Mô tả lại mục đích của test case và kết quả thực tế không như kết quả mong đợi
- ❑ Ví dụ:
  - Không thông báo lỗi khi đăng kí tài khoản đã tồn tại
  - Không thông báo lỗi khi ô email bỏ trống
  - Không thông báo lỗi khi số khách quá số lượng qui định
  - Tiền phòng tính sai trong trường ngày đi bằng ngày đến

# 4. How to reproduce it

- ❑ Mô tả các bước để tái hiện lại lỗi
- ❑ Dừng lại test step, expected result, actual result
- ❑ Ví dụ:
  1. Vào chức năng tính tiền phòng
  2. Nhập [Ngày đến] là “1/1/2015”
  3. Nhập [Ngày đi] là “1/1/2015”
  4. Chọn [Loại phòng] là A
  5. Nhập [Số khách] là 1
  6. Nhập [Tiền dịch vụ] là 0
  7. Bấm nút [Tính tiền]
    - Kết quả mong đợi: 450; kết quả thực tế: 0
- ❑ Các thông số về môi trường khi chạy test case

# 5-8. Nội dung báo cáo lỗi

## 1. Reported by

- Người tạo ra báo cáo lỗi

## 2. Date

- Ngày tạo báo cáo

## 3. Assign to

- Phân công developer sửa lỗi

## 4. Status: trạng thái lỗi

- Ban đầu: Open
- Sau khi xử lý: Fixed/Resolved
- Không ảnh hưởng nữa: Closed

# 9. Priority

- ❑ Độ ưu tiên xử lí lỗi
- ❑ Phân loại
  - Fix immediately
  - Fix as soon as possible
  - Must fix before the next milestone
  - Must fix before final
  - Fix if possible
  - Optional

# 10. Severity

## ☐ Độ nghiêm trọng của lỗi

## ☐ Phân loại

### ■ Minor

Chưa test invalid data, giá trị biên, chạy chậm, layout ko tốt, tab order sai, default focus sai, short key sai

### ■ Serious

Lỗi làm cho chức năng chính của hệ thống ko hoạt động được, hoặc hoạt động sai

### ■ Fatal

Lỗi cực kì nghiêm trọng, làm hệ thống crash, ko thể phục hồi dữ liệu

# Đặc điểm của bug report

- ❑ Written – Được viết ra
  - Mô tả lỗi bằng văn bản
  - Phải có báo cáo lỗi, dù lỗi có thể sửa nhanh
- ❑ Numbered – Được đánh chỉ mục
  - Truy vết được thông qua chỉ mục
- ❑ Simple – Đơn giản
  - 1 lỗi → 1 báo cáo
  - Nhiều lỗi trên 1 báo cáo: dễ gây nhầm lẫn
- ❑ Understandable – Dễ hiểu
  - Mô tả lỗi rõ ràng
  - Bỏ các chi tiết ko cần thiết

# Đặc điểm của bug report

- ❑ Reproducible – Có thể tái tạo lỗi
  - Khách hàng thường báo cáo lỗi ko thể tái tạo
  - Mô tả từng bước rõ ràng, lưu ý trạng thái ban đầu
  - Phân tích để chọn số bước ít nhất
- ❑ Legible – Rõ ràng
  - Áp dụng cho báo cáo tay
- ❑ Non-judgmental – Không phán xét
  - Ko thể hiện cảm xúc cá nhân
  - Mâu thuẫn có thể gây hậu quả nghiêm trọng

# Tạo một lỗi có thể tái tạo

- ☐ Ghi lại mọi thứ đã làm trong lúc kiểm thử
- ☐ Sử dụng chương trình ghi lại thao tác chuột, bàn phím
- ☐ Sử dụng chương trình quay màn hình



# Test summary report

- ❑ Tài liệu tóm tắt hoạt động kiểm thử và kết quả
- ❑ Cấu trúc
  - Summary
  - Test case result report
  - Defect report
  - Open point

# Test case result report

## ❑ Thống kê lỗi theo chức năng

### TEST REPORT

<b>Project name</b>	<Project name>	<b>Reviewer</b>	<Reviewer>
<b>Creator</b>	<Creator>	<b>Approver</b>	<Approver>
<b>Note</b>			

Test Coverage: **46%**  
Successful Test Coverage: **33%**

Date: <yyyy/mm/dd>

No	Items	Tested	Passed	Failed	Blocked	Skipped	Not Yet Tested	Total	Tested Coverage
1	Function 1	23	15	5	3	7	18	48	48%
2	Function 2	26	20	4	2	10	22	58	45%
3									
4									
5									
Total		49	35	9	5	17	40	106	

# Test metrics

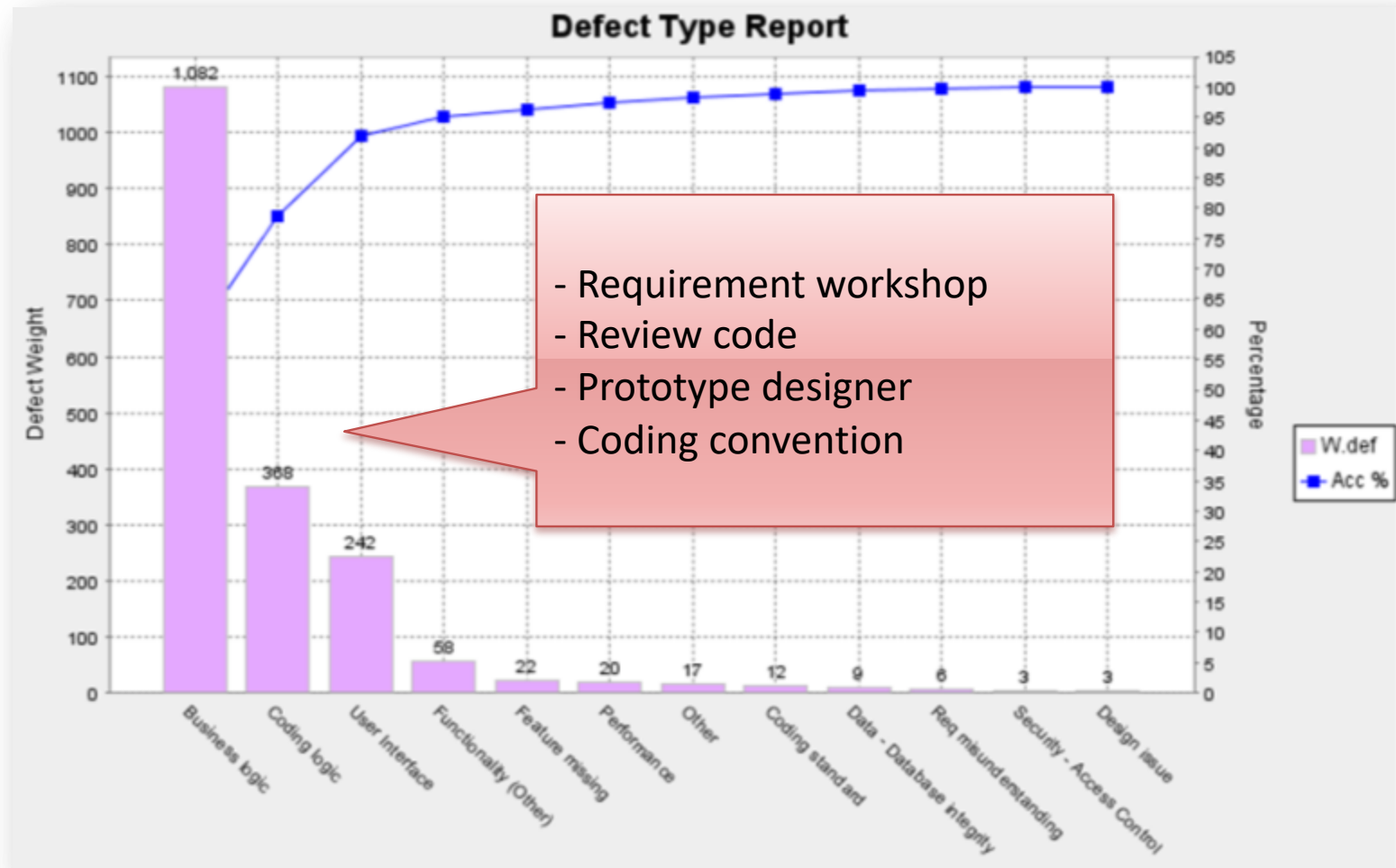
- ❑ Tested (No of Executed Test Case) = Passed + Failed + Blocked
- ❑ Test Coverage: tỉ lệ phần trăm của No of Executed Test Case và Total Test Case
- ❑ Test Success Converage: tỉ lệ phần trăm của No of Passed Test Case và No of Executed Test Case

# Defect type report

- Thống kê lỗi theo phân loại lỗi, cho thấy loại lỗi hay xuất hiện

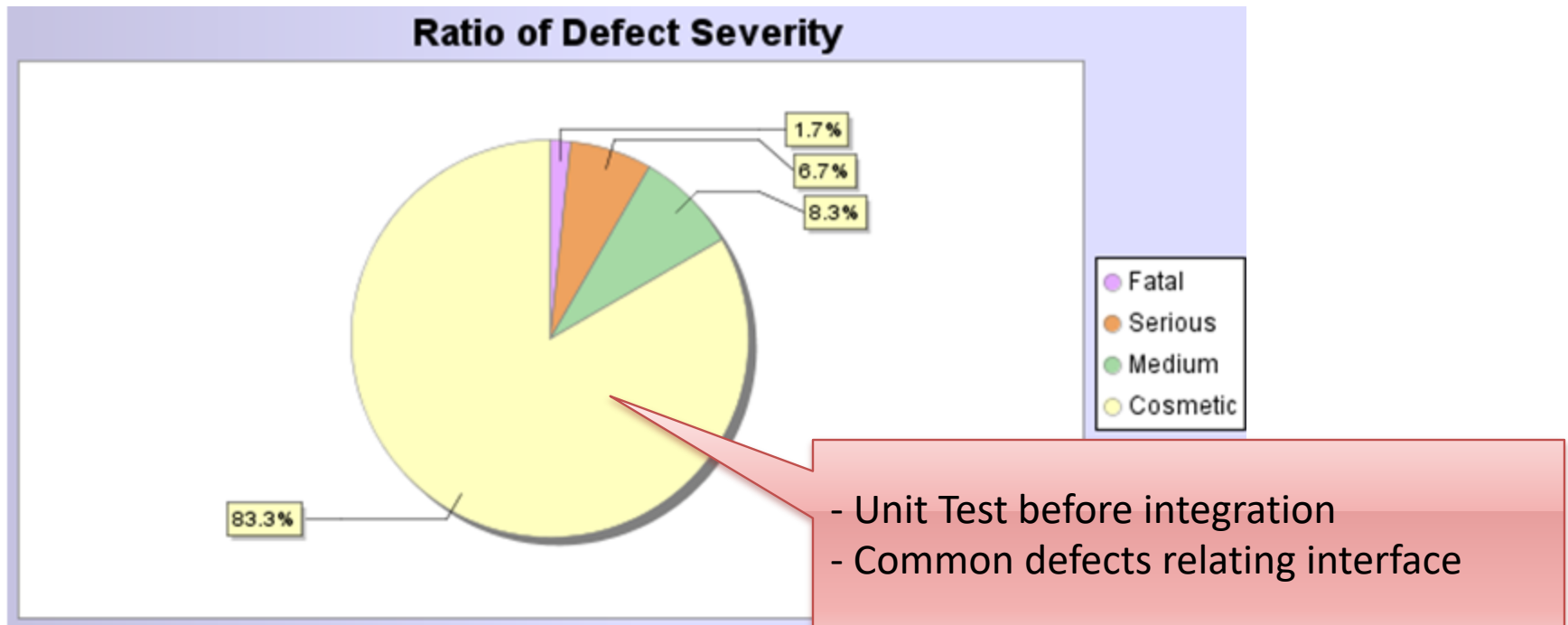
Defect Type	Fatal	Serious	Medium	Cosmetic	Total (W.def)	%
Business logic	1	9	332	31	1082	58.7
Coding logic	1	2	112	12	368	20
Coding standard			4		12	0.7
Data - Database integrity			3		9	0.5
Design issue			1		3	0.2
Feature missing			6	4	22	1.2
Functionality (Other)			18	4	58	3.1
Other			5	2	17	0.9
Performance		2	3	1	20	1.1
Req misunderstanding			2		6	0.3
Security - Access Control			1		3	0.2
User Interface			30	152	242	13.1
<b>Total</b>	<b>2</b>	<b>13</b>	<b>517</b>	<b>206</b>	<b>1842</b>	<b>100</b>

# Defect type report



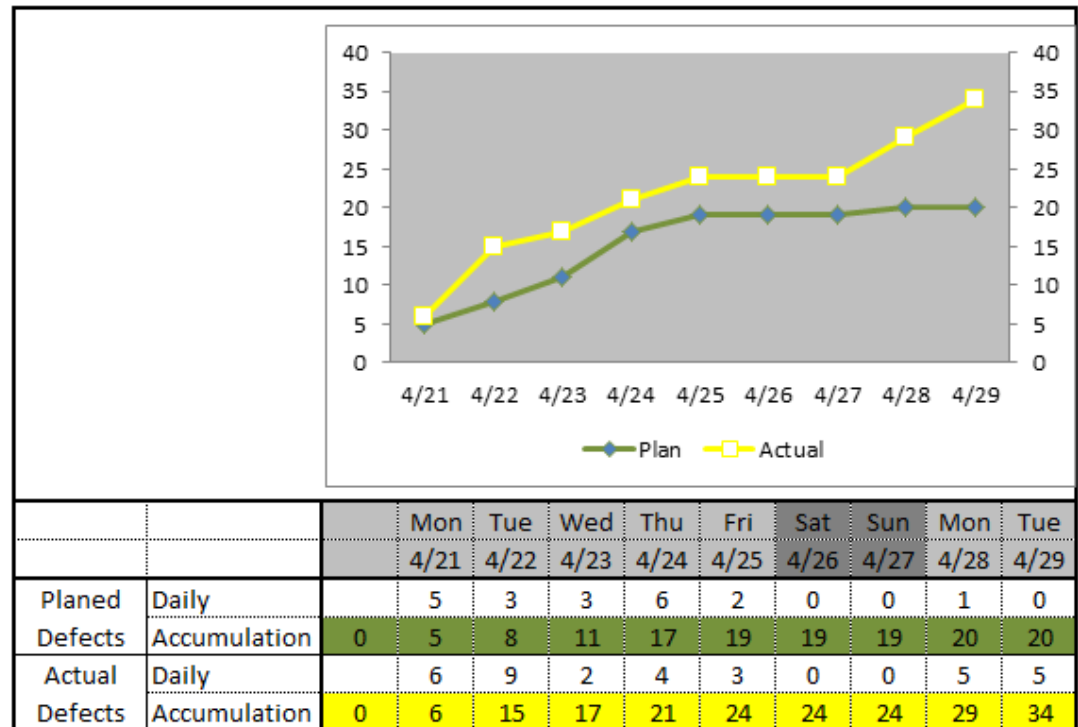
# Defect severity report

- Thống kê lỗi theo độ nghiêm trọng
  - Nhận xét: đa số lỗi là về giao diện

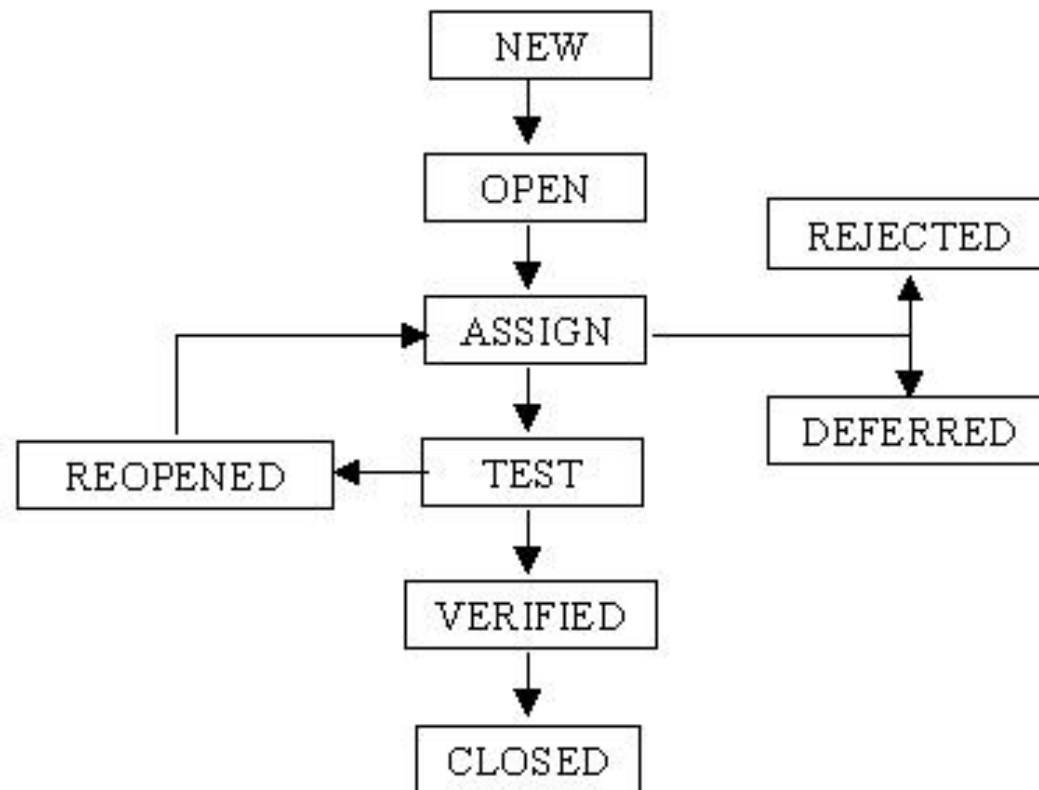


# Defect trend report

- Thống kê khuynh hướng lỗi, dự đoán lỗi ở tương lai gần



# Vòng đời của lỗi phần mềm





# Vòng đời của lỗi phần mềm

- ❑ NEW: Khi defect được log và post lần đầu
- ❑ ASSIGNED: tester post defect → lead of tester approve → assign bug cho developer tương ứng
- ❑ OPEN: developer bắt đầu phân tích và xử lý bug
- ❑ FIXED: developer đã xử lý xong, báo lại và chuyển giao cho testing team
- ❑ PENDING RETEST: đợi tester retest
- ❑ RETEST: tester retest code thay đổi và báo lại kết quả cho developer

# Vòng đời của lỗi phần mềm

- ❑ VERIFIED: tester retest → bug đã hết → tester approve
- ❑ REOPEN: tester retest → bug vẫn còn
- ❑ CLOSED
- ❑ DUPLICATED: khi 1 bug lặp lại 2 lần hoặc 2 bug cùng liên quan đến 1 concept
- ❑ REJECTED: developer nghĩ ko phải là bug
- ❑ DEFERRED: bug sẽ được fix trong đợt release tiếp theo. Lí do: priority thấp, thiếu thời gian, bug ko gây ra hậu quả nghiêm trọng cho hệ thống
- ❑ NOT A BUG: chẳng hạn, khách hàng yêu cầu thay đổi look-and-feel

# Thuật ngữ

- ❑ Test case execution status
  - PASSED, FAILED, BLOCKED, SKIPPED, NOT YET TESTED
- ❑ Bug report – Defect report
  - Bug ID
  - Reproducible
  - Bug priority
  - Bug severity
- ❑ Test summary report
  - Test case result report
  - Test metrics: test coverage, test success coverage
  - Defect type report
  - Defect severity report
  - Defect trend report
- ❑ Bug life cycle