

# Test Execution & Report



Global CyberSoft

*A World of Difference*

By **Truong Ho**

*July, 2012*

*Duration: 3 hours*

[www.globalcybersoft.com](http://www.globalcybersoft.com)

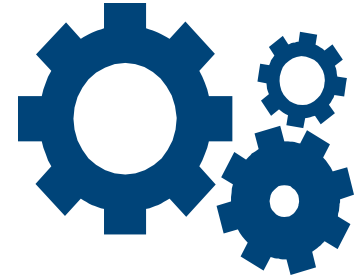


**Microsoft**  
**GOLD CERTIFIED**  
*Partner*

# Contents

---

1. Scope and purpose of this course ?
2. Test case management and execution
3. Bug report
4. Bug tracking system
5. Test result report
6. Roles & Responsibilities
7. Skills



# 1. Scope and purpose

---

## □ Focus around 3 main objects

- ⌘ Test case

- ⌘ Bug

- ⌘ Specification / Design

## □ How-to

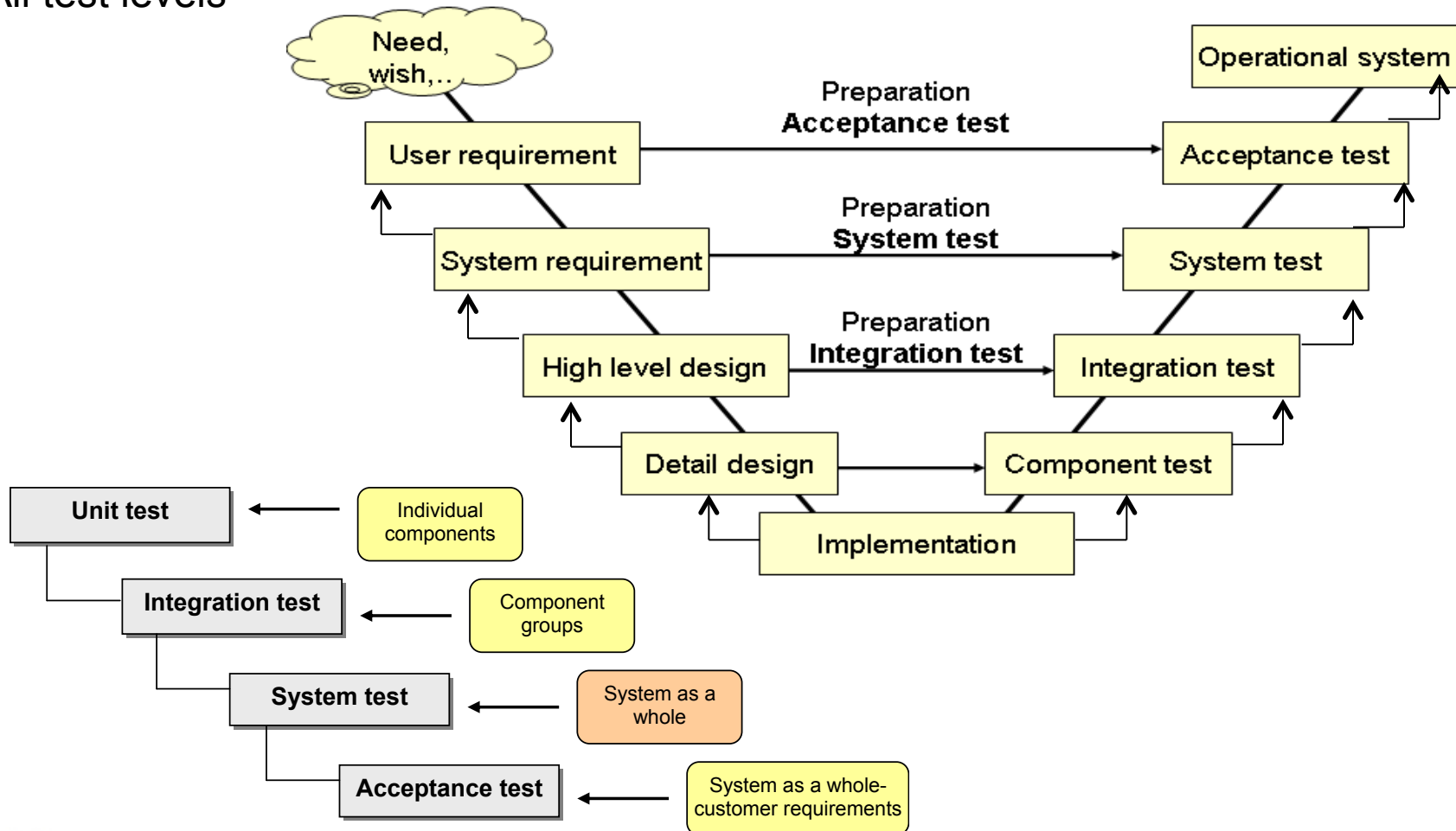
Manage ⇒ Retrieve ⇒ Manipulate ⇒ Report ⇒ Review



# 1. Scope and purpose (cont)

## □ Phases on V-Model:

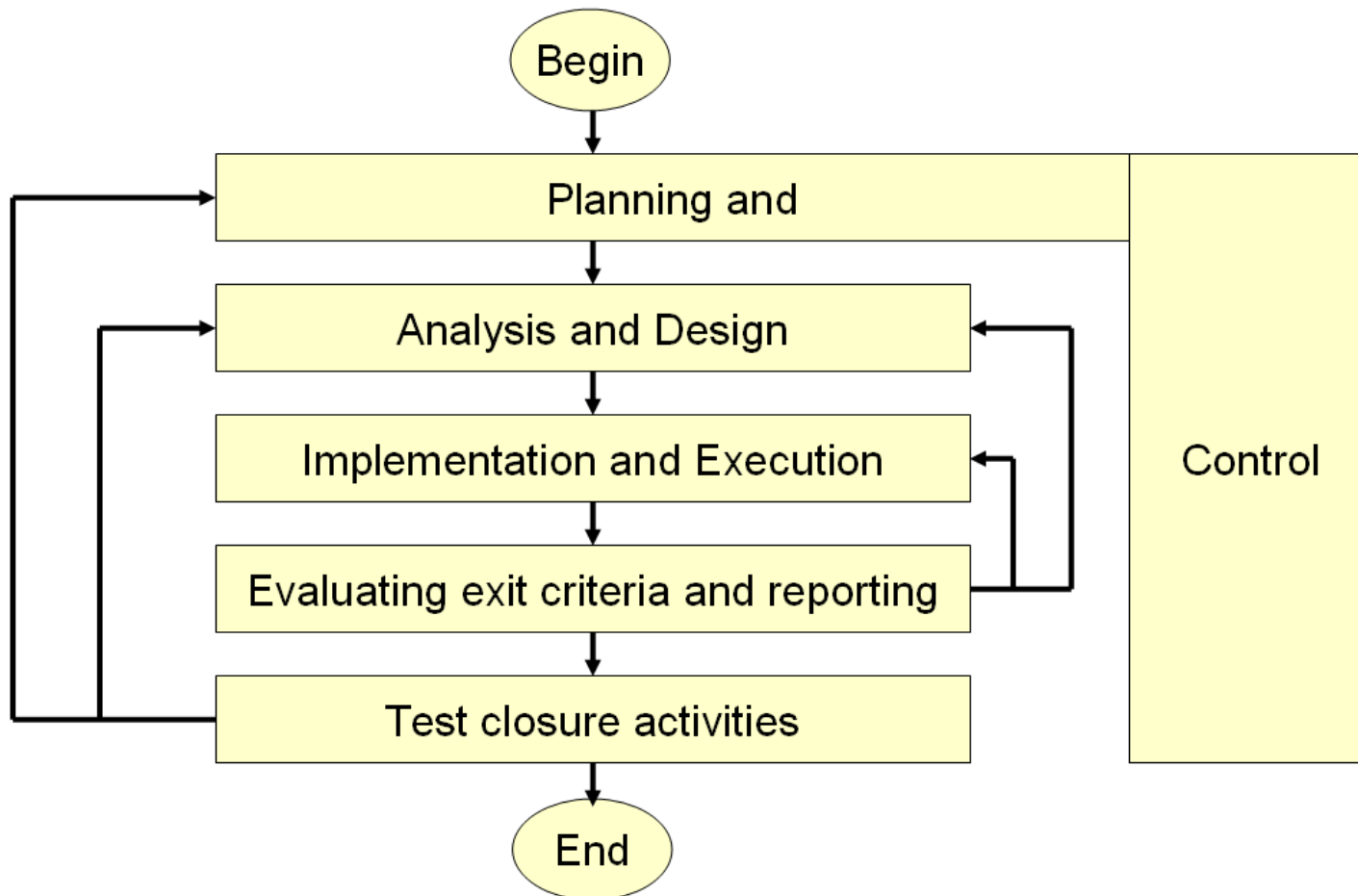
All test levels



# 1. Scope and purpose (cont)

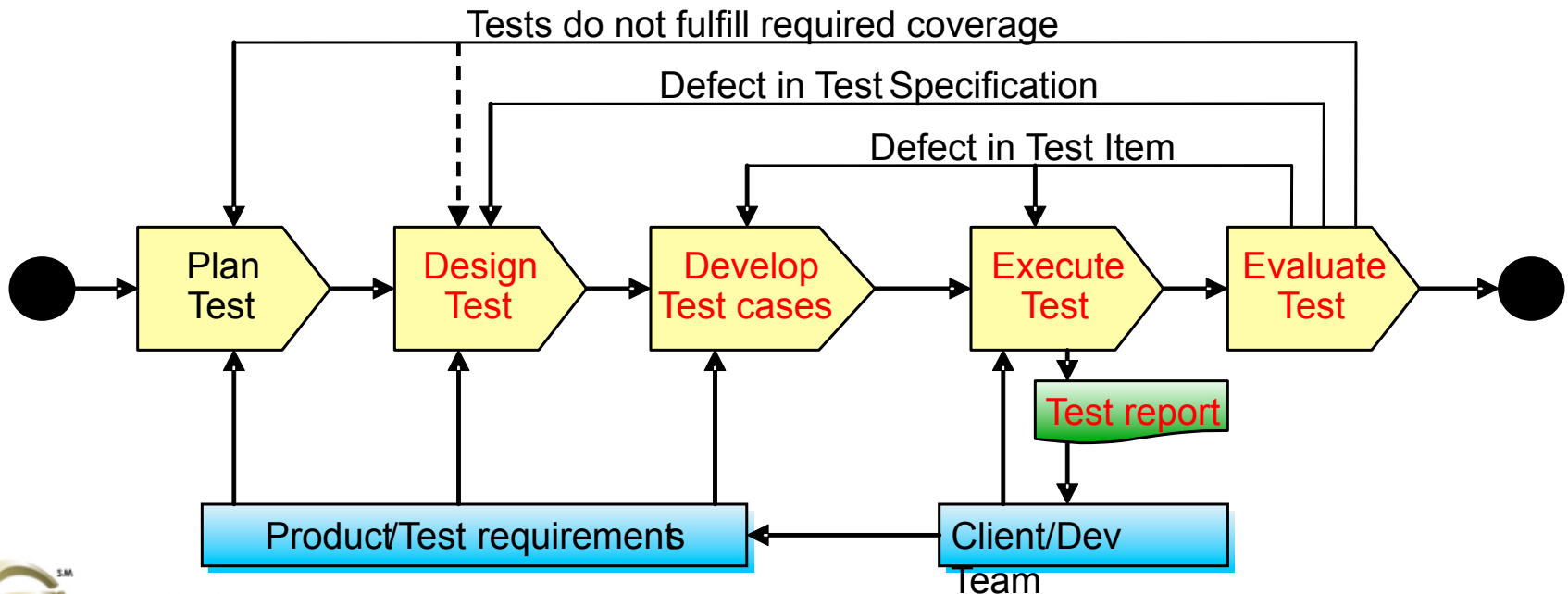
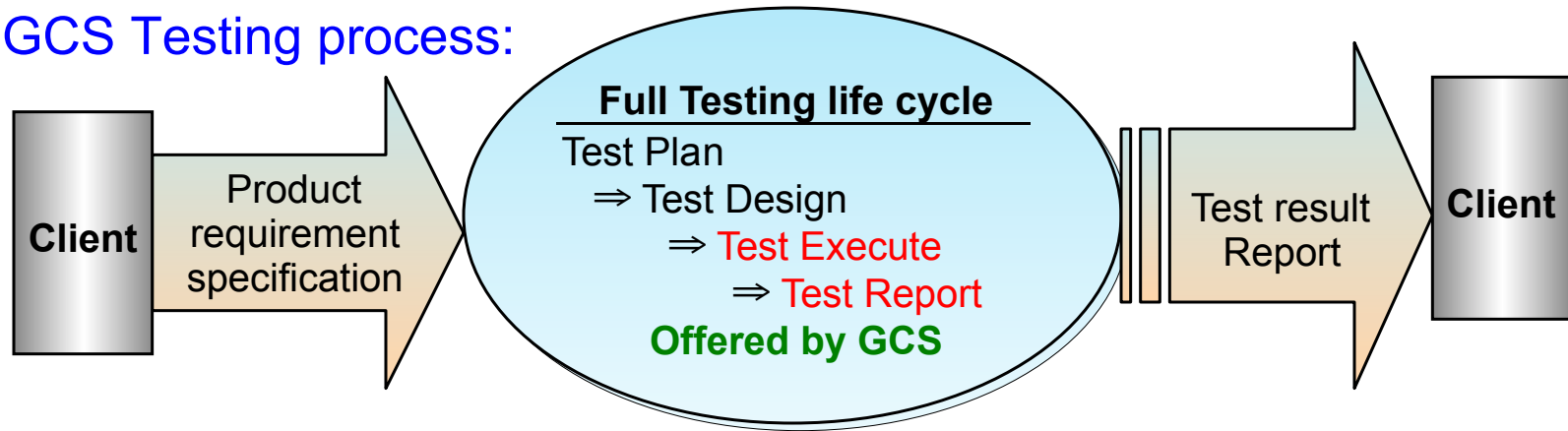
## □ Phases on Testing process:

Execution ➡ Evaluating exit criteria & report



# 1. Scope and purpose (cont)

## □ GCS Testing process:

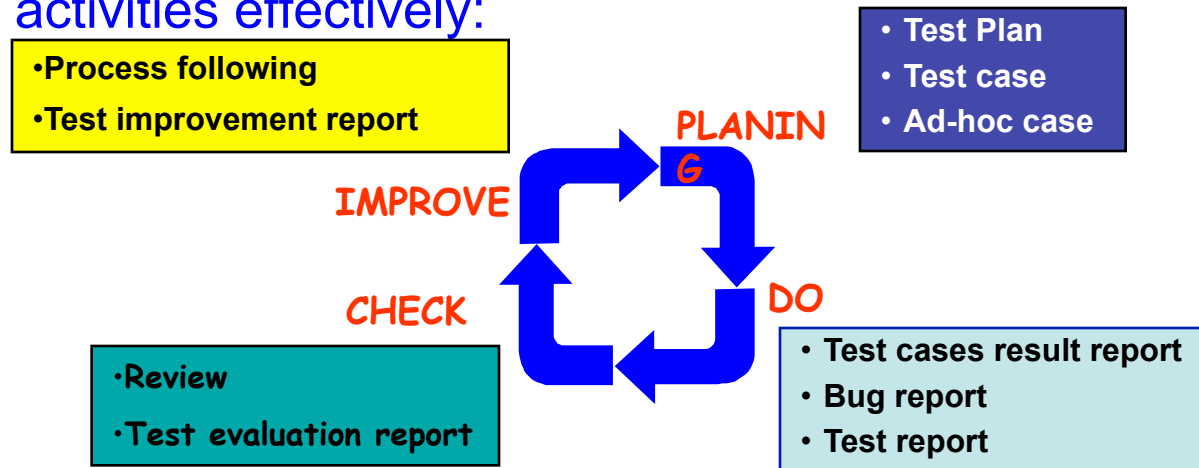


# 1. Scope and purpose (cont)

## □ Activity flow:

- ⇒ Setup test environment
  - ⇒ Manage and retrieve tests
  - ⇒ Execute test cases & Fill test case results
  - ⇒ Report bug
    - ⇒ Report improvement (if any)
    - ⇒ Report test results

## □ How to do activities effectively:



## 2. Test case management & execution

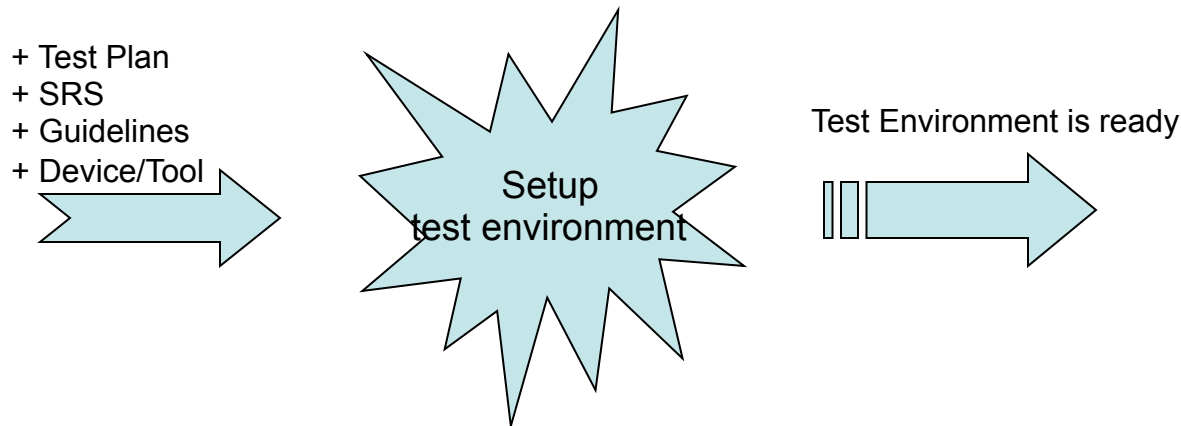
### 2.1 Setup test environment:

#### ✓ Input:

Test plan ; Software Requirement Specifications (SRS) ; Guidelines ; Templates ; Devices/Tools ; etc. as defined in Test Plan or Test Strategy

#### ✓ Output:

Checklist/Criteria for validating test environment (as declared in Test plan) ; And the established and configured test environment





## 2. Test case management & execution

---

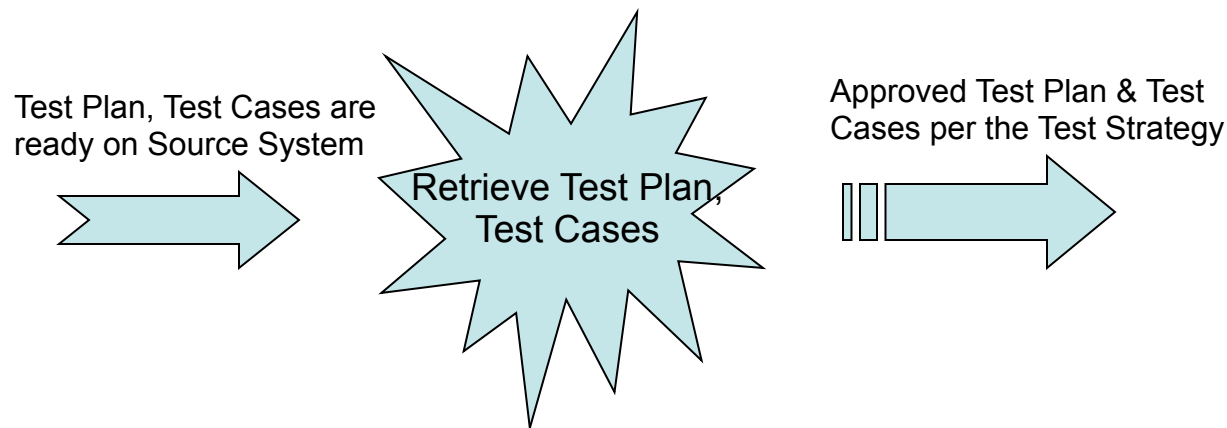
### 2.2 Manage and retrieve tests:

✓ **Input:**

Source System to store Test plan, Test cases

✓ **Output:**

Source system to store the Test Plan, Test cases



## 2. Test case management & execution

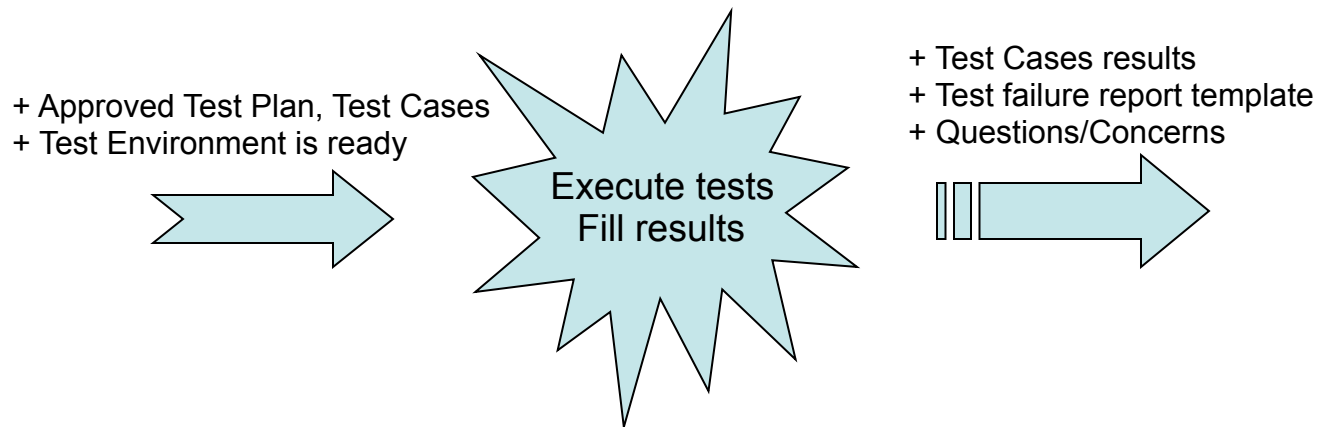
### 2.3 Execute test cases & fill test case results:

#### ☑ Input:

Approved Test plan, Test scenarios, Test cases, or Test Harness  
Test Environment is ready

#### ☑ Output:

Test case results to result sheets or Source system  
Test failure report template  
Question/Concern list (if any)



## 2. Test case management & execution (cont)

---

- ❑ **Manual = Auto**
  
- ❑ **Some test case status:**
  - Not Run
  - In-Progress
  - Passed
  - Failed Existing
  - Failed Regression
  - Failed New
  - Blocked

### 3. Bug report

---

#### ☑ Input:

Test case failures

Corner case failures

#### ☑ Output:

Bug posted to Bug Tracking System

Testcase/Spec/Feature improvement appendix (if any)

Question/Concern list (if any)

+ Test case failure  
+ Corner case failures



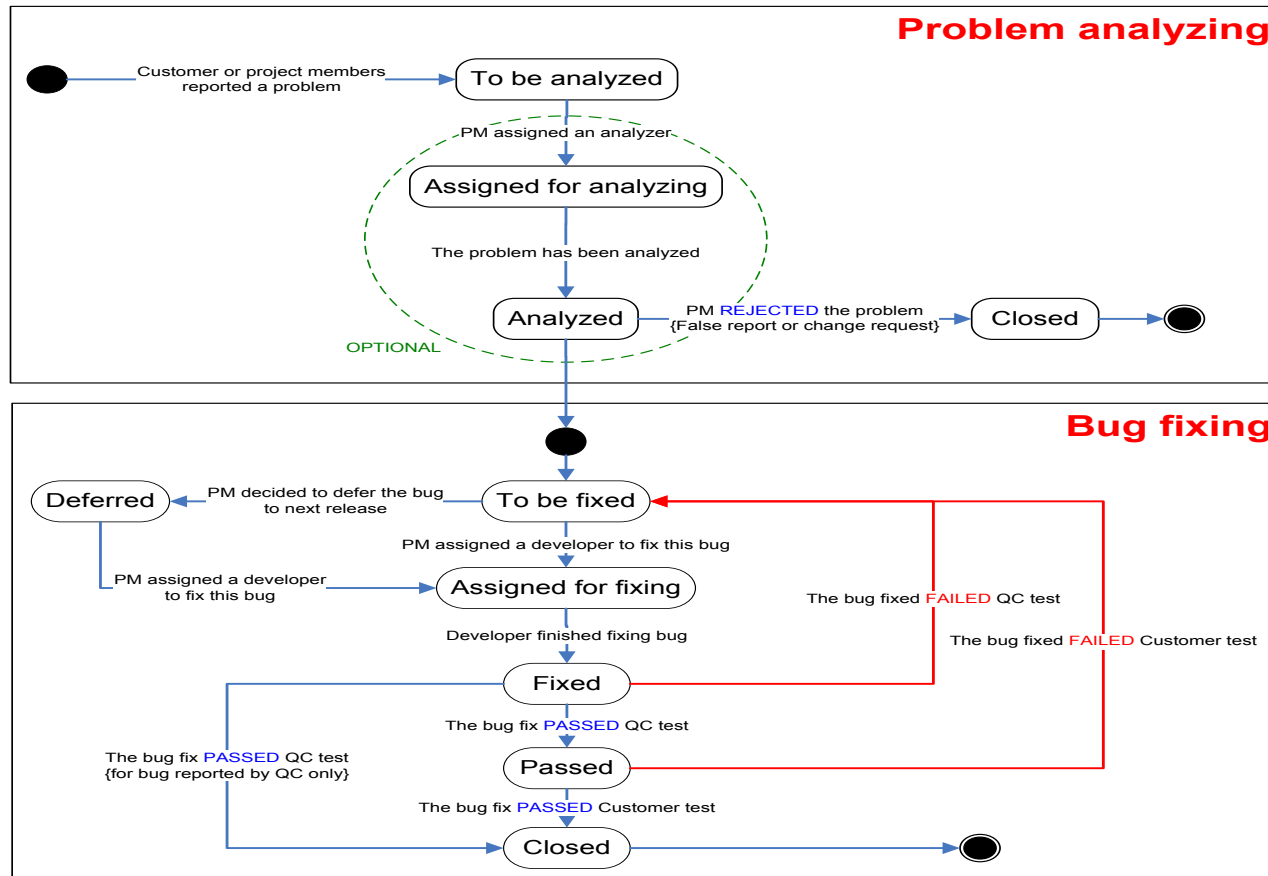
+ Bug posted to Bug Tracking System  
+ Test Improvement request  
+ Questions/Concerns



### 3. Bug report (cont)

#### □ Activity flow (tailored to cover Bug's Life Cycle)

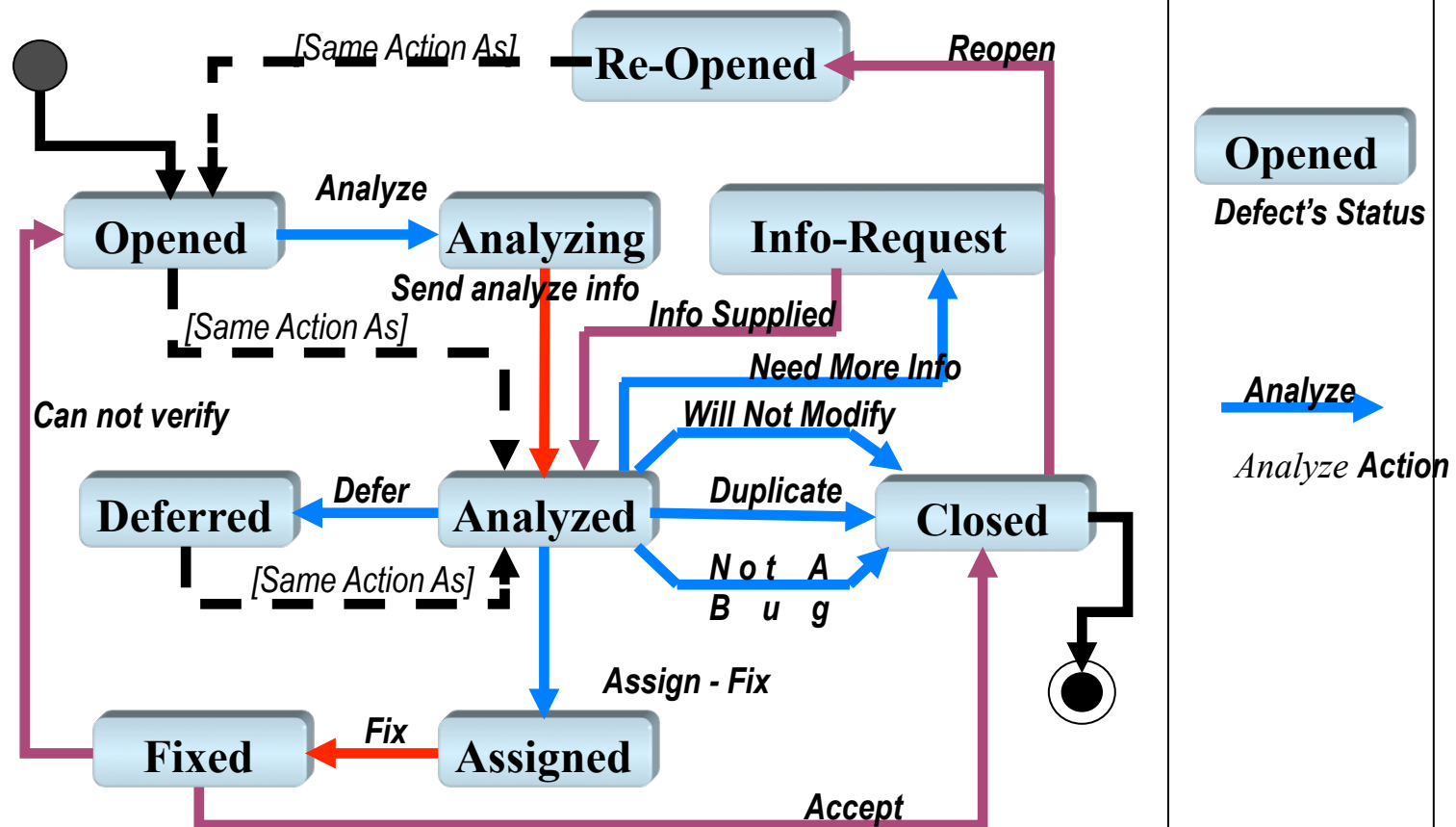
Bug ⇒ Local review ⇒ Assign ⇒ Analyze/Resolve ⇒ Route to bug fixer ⇒ Fix/Resolve ⇒ Verify ⇒ Close/Re-open



### 3. Bug report (cont)

#### □ Activity flow (tailored to cover Bug's Life Cycle)

Bug ⇒ Local review ⇒ Assign ⇒ Analyze/Resolve ⇒ Route to bug fixer ⇒ Fix/Resolve ⇒ Verify ⇒ Close/Re-open



### 3. Bug report (cont)

---

#### ❑ Some Resolution on Resolve activity

- ⌚ Unresolved (default)
- ⌚ Need Info
- ⌚ Invalid
- ⌚ Duplicated
- ⌚ Won't Fix
- ⌚ Defer (on-hold)
- ⌚ Cannot reproduce
- ⌚ Incomplete

#### ❑ Bug's Life Cycle = Status (Open, In-Progress, Resolved, Re-opened, Closed, On-Hold, Delivered, Patch Ready, In Testing)

### 3. Bug report (cont)

---

#### □ Some bug types:

- ⊕ Test case's bug
- ⊕ Ad-hoc bug
- ⊕ Regression bug
- ⊕ Enhancement bug

#### □ Some bug attributes:

- ⊕ Reproducibility
- ⊕ Customer Impact
- ⊕ Severity vs Priority (Tester vs Developer)



### 3. Bug report (cont)

Severity	Definition	Typical Action
<b>0: Critical</b>	Device/System/Software does <b>NOT work at all</b>	<ul style="list-style-type: none"> <li>Review test cases/scenarios</li> <li>Inform dev team/client immediately</li> <li>Analyze the root cause when necessary</li> </ul>
<b>1: Major</b>	<b>Major</b> function disabled or incorrect	<ul style="list-style-type: none"> <li>Review test cases/scenarios</li> <li>Inform dev team/client ASAP</li> <li>Analyze the root cause when necessary</li> </ul>
<b>2: Minor</b>	Some inconvenience for users	<ul style="list-style-type: none"> <li>Record and track in bug tracking</li> </ul>
<b>3: Cosmetic</b>	Spelling errors etc., no effect on operation	<ul style="list-style-type: none"> <li>Record and track when necessary</li> </ul>

Priority	Definition
<b>0: Emergency</b>	<b>Work must stop until this bug is closed.</b>
<b>1: High</b>	<b>Work is impeded while this bug is open</b>
<b>2: Medium</b>	<b>Some negative impact on work exists until this bug is closed</b>
<b>3: Low</b>	<b>This bug must be closed, but it does not interfere with current work.</b>

### 3. Bug report (cont)

---

#### □ Bug report template

##### **Header**

- Summary
- Severity
- Affected build/device-type
- Tested environment
- Reproducibility
- Bug type
- Customer Impact
- Recovery steps

### 3. Bug report (cont)

---

#### □ Bug report template

##### **Body/Description**

- Steps to reproduce
- Expected result
- Actual result
- System logs
- Tested Functional area
- Screenshot (if any)
- Notes

## 4. Bug tracking system

---

### □ General requirements:

- ⊕ Log and track Defect's life cycle, Defect's analyze/resolve action
- ⊕ View responsibility of parties
- ⊕ Search & filters
- ⊕ Report

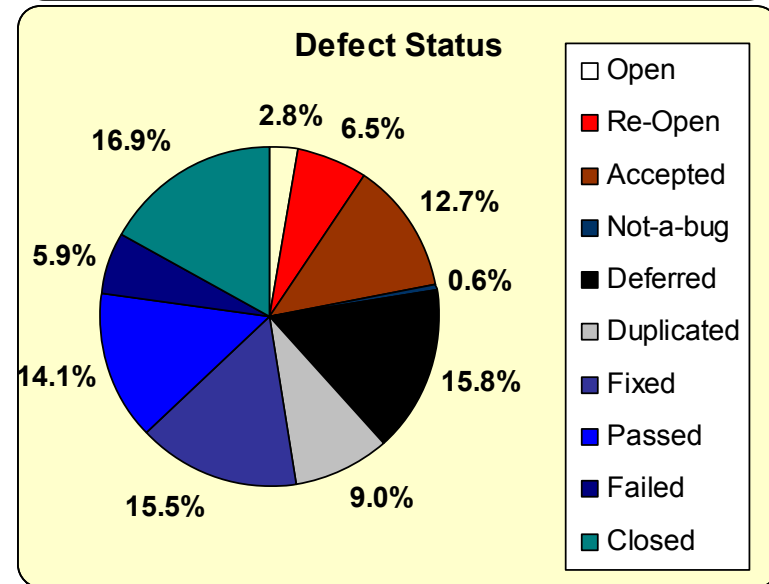
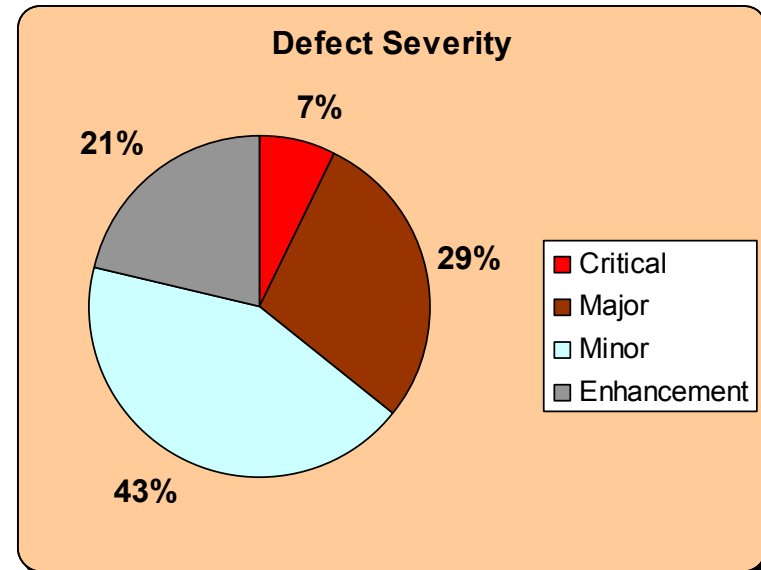
### □ Some popular systems:

- ⊕ Bug Zilla
- ⊕ JIRA
- ⊕ Clarify ClearQuality
- ⊕ ClearQuest

## 4. Bug tracking system (cont)

### □ Demo report

GENERAL SUMMARY		
Total	558	
Status	#	%
Open	0	0.0%
Re-Open	0	0.0%
Accepted	0	0.0%
Not-a-bug	37	6.6%
Deferred	12	2.2%
Duplicated	1	0.2%
Fixed	0	0.0%
Passed	0	0.0%
Failed	0	0.0%
Closed	508	91.0%
Severity	#	%
Critical	1	0.2%
Major	347	67.0%
Minor	170	32.8%
Enhancement	0	0.0%



## 5. Test result report

### ☑ Input:

Filled Test cases with results  
Bug tracking list attached  
Improvement request  
Question/Concern list

### ☑ Output:

Test result report  
Bug statistic and life cycle report

- + Filled Test cases with results
- + Bug tracking list
- + Question, Concerns list
- + Improvement request



- + Test result report (test case/bug results and progress)
- + Test statistics report (test case and bug statistics)
- + Questions/Concerns, Improvement appendix



# 5. Test result report (cont)

## □ Demo report

### GENERAL SUMMARY

Total	0	
Status	#	%
Open	0	0.0%
Re-Open	0	0.0%
Accepted	0	0.0%
Not-a-bug	0	0.0%
Deferred	0	0.0%
Duplicated	0	0.0%
Fixed	0	0.0%
Passed	0	0.0%
Failed	0	0.0%
Closed	0	0.0%
Severity	#	%
Critical	0	0.0%
Major	0	0.0%
Minor	0	0.0%
Enhancement	0	0.0%

Total Test Cases: 36

### TEST RESULT

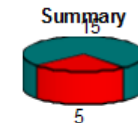
Executed	36	100.0%
High	27	75.0%
Medium	9	25.0%
Low	0	0.0%
Passed	36	100.0%
High	27	75.0%
Medium	9	25.0%
Low	0	0.0%
Failed	0	0.0%
High	0	0.0%
Medium	0	0.0%
Low	0	0.0%



### TEST EXECUTION REPORT

Information	
Project name	
PM	
Reported By	
Date	

Test Result Summary			
#Test Cases	58		
#Executed	20	34.5%	
#Passed	15	75.0%	
#Failed	5	25.0%	



Test Case ID	Test Cases Description	Priority	Result	Tester	Build 060619-05 Status Date	Build 060614-00 Status Date	Build 060614-00 Status Date	Note
Function 01								
FT-LOGIN-01	Test case <a href="#">login</a> 01							
FT-LOGIN-02	Test case <a href="#">login</a> 02							
FT-LOGIN-03	Test case <a href="#">login</a> 03		P		P	F	F	
FT-LOGIN-04	Test case <a href="#">login</a> 04		P				P	
FT-LOGIN-05	Test case <a href="#">login</a> 05		P		I	P		
FT-LOGIN-06	Test case <a href="#">login</a> 06							
FT-LOGIN-07	Test case <a href="#">login</a> 07							
FT-LOGIN-08	Test case <a href="#">login</a> 08							
FT-LOGIN-09	Test case <a href="#">login</a> 09							
Function 02								
FT-LOGIN-01	Test case <a href="#">login</a> 01							
FT-LOGIN-02	Test case <a href="#">login</a> 02							
FT-LOGIN-08	Test case <a href="#">login</a> 08							
FT-LOGIN-09	Test case <a href="#">login</a> 09							



## 6. Roles & Responsibilities

- × **Setup Test Environment using criteria**

Tester      ⇒ Test Lead

- × **Manage test plan, test case**

Test Lead    ⇒ PM

- × **Execute Test and report test case result**

Tester      ⇒ Test Lead

- × **Report bugs**

Tester      ⇒ Test Lead

- × **Report Questions/Concerns**

Tester      ⇒ Test Lead

- × **Improvement request**

Tester      ⇒ Test Lead

- × **Report test result**

Test Lead    ⇒ PM

Tester      ⇒ Test Lead (optional on some specific projects)





## 7. Skills

### ❑ Test case result report skill

**Understand correctly what status the test case is**

- ➡ When is PASSED ?
- ➡ When is FAILED ?
- ➡ When is IN-PROGRESS ?
- ➡ When is BLOCKED ?



**Understand correctly if a failure is test case bug or corner bug**

**How to overcome unclear test cases & test execution conflicts**

- ✓ What we can do ?
- ✓ When we should do ?
- ✓ How we do ?

~~~~~ See you in **“Test Maintenance”** topic ~~~~~

## 7. Skills (cont)

### □ Bug report skill

#### Writing summary

- ◆ One-line summary (will not be more than 50-60 characters)
- ◆ Symptom + Action + Condition (SAC)
- ◆ Make the symptom as much critical as possible, but exactly the issue



#### Writing description & steps to reproduce

- ◆ Steps are always MAXIMUM REDUCED to meet the issue
- ◆ Steps by number. Combine simple steps by ">"
- ◆ Re-test steps to reproduce by another tester
- ◆ Notes are always useful to simplify the steps, and to provide full information
- ◆ Always have a second chance to reproduce the issue

#### Writing results

- ◆ Expected = Actual (for all same things)  
Use as many same words/sentence as possible on both Expected & Actual results  
Just the difference is different → easy to see what the issue is
- ◆ Summary = Actual result

#### E.g.

E = The mobile phone is turned on then welcome screen is loaded with graphics

A = The mobile phone is turned on but welcome screen is NOT loaded, just gray screen appears

S = Gray screen and no welcome screen loaded after turned on the mobile phone

## 7. Skills (cont)



### □ Bug report skill

#### Keep all things **CORRECT**

- ✚ Double-check the links/attachments: log, screenshot, reference issue
  - **Correct bug data**
- ✚ Test environment & pre-requisites are always clear
  - **Correct test data**
- ✚ If bug reproducibility is low, try to keep the test environment unchanged until getting further information from developer. Should hold on any information regarding the bug
  - **Correctly a bug. Protect your bug's reliability**
- ✚ **Language**
  - ✓ Keep It Simple & Straight. You are not writing an essay or an article, so use simple language
  - ✓ Focus on the issue/problem; not the person
  - ✓ Always respect Developer
  - **Correct communication**

## 7. Skills (cont)

### □ Notes in mind

- The less re-work the quicker bug fix
  - Full information
- The more unclear, the bug won't get fixed sooner
  - Clear & Correct bug
- The longer description, the more chance to be postponed
  - Simple description, easy issue understanding
- Report for one issue, not some issues
  - One report one single issue
- The more comparison checking result the more useful bug report
  - Useful information makes developer happy
- Good tester is not for bug, is for a bug root cause elimination
  - Try to find the root cause, or at least show investigation steps



# Questions

---



# Practices/ Exercises

---



*A World of Difference*

# Reference

---

□ Refer to:

- **GCS Testing Foundation (Son Pham)**
- **Basic Testing Techniques (Son Pham)**
- **Test Case Workshop (Son Pham)**

# Appendix: Course detail form

---

|                 |                         |                 |                     |
|-----------------|-------------------------|-----------------|---------------------|
| <b>Author</b>   | Truong Ho               | <b>Duration</b> | 3 hours             |
| <b>Category</b> | Test Execution & Report | <b>Type</b>     | Theory and Practice |

|                                               |                                                                          |
|-----------------------------------------------|--------------------------------------------------------------------------|
| <b>Examination</b>                            | N/A                                                                      |
| <b>Intended Audience</b>                      | Any QC                                                                   |
| <b>Pre-requisites</b>                         | N/A                                                                      |
| <b>Completion criteria for the course</b>     | Attendee must join at least 90% course length                            |
| <b>Criteria for granting training waivers</b> | Those who has experience on Test Case Design area or ISTQB certification |



# Thank you

---

## THANK YOU

Inquires regarding the above may be directed to:  
**Someone, Title,** [truonghx@gcs-vn.com](mailto:truonghx@gcs-vn.com)