

Defect Reporting and Defect Life Cycle Management

Lesson 00

People matter, results count.



©2016 Capgemini. All rights reserved.

The information contained in this document is proprietary and confidential. For Capgemini only.

Document History

Date	Course Version No.	Software Version No.	Developer / SME	Reviewer(s)	Approver	Change Record Remarks
	0.1D	NA				Content Creation
	0.1	NA				Review
May-2009	1.0	NA	Priya Rane			Material Revamp
June-2011	1.1	NA	Kishori Khadiolkar			Material Revamp
June-2015	1.2	NA	Alphy Thomson	Gurpreetkaur Marve	Shilpa Bhosle	Material Revamp

Course Goals and Non Goals

- Course Goals

- At the end of this program, participants gain an understanding of how to log the defect in the defect tracking tool/sheet and how to create the defect free defective defect report.

- Course Non Goals

- This course does not cover automation process of testing



Pre-requisites

- The participants have attended the training on Software Testing
- The participants can understand and interpret Use Cases/have attended the “Requirements Development/Requirements Management” training covering Use Cases

Intended Audience

- Test Engineers and Senior Test Engineers



Day Wise Schedule

- Day 1
 - Lesson 1: Defect Free Defect Reporting

Table of Contents

- Lesson 1: Defect Free Defect Reporting
 - 1.1 What is Defect and Defect Report
 - 1.2 Need of Defect reports
 - 1.3 Defect Report Template
 - 1.4 Important Attributes of a Defect Report
 - 1.5 Defect Life Cycle
 - 1.6 Users for Defect Report
 - 1.7 What is “Defective defect report”
 - 1.8 Reasons for defective reports
 - 1.9 Root causes and other influencing factors for defective reports

Table of Contents

- Lesson 1: Defect Free Defect Reporting (Contd..)
 - 1.10 Defect Free Reports
 - 1.11 Defective reports – Certain facts
 - 1.12 Writing Defect Free Reports
 - 1.13 Reporting and communication Process
 - 1.14 Writing Effecting Defect Reports
 - 1.15 Using Tools for Reporting
 - 1.16 Project / Organization Level Process
 - 1.17 Defect Free Reports – Advantage
 - 1.18 Nine commitments

References

- Student material:
 - Class Book (presentation slides with notes)



Next Step Courses (if applicable)

- Defect Tracking Tool - Bugzilla



Defect Reporting and Defect Life Cycle Management

Lesson1: Defect Free Defect
Reporting

Lesson Objectives

- To understand the following topics:
 - What is Defect and Defect Report
 - Need of Defect reports
 - Defect Report Template
 - Important Attributes of a Defect Report
 - Defect Life Cycle
 - Users for Defect Report
 - What is “Defective defect report”
 - Reasons for defective reports
 - Root causes and other influencing factors for defective reports



Lesson Objectives

- To understand the following topics:
 - Recommendations for defect free reports
 - Some Realities about defective reports
 - Severe Defect
 - Preparation for writing defect free reports
 - Reporting and Communication of defects
 - Advantages of using tool for Reporting
 - Project / Organization level Process
 - Advantages of defect free reports
 - Nine Commitments



1.1: Defect and Defect Report

Definition

- Defect
 - If AUT's (Application Under Test's) some feature or function is not working as per what is there in requirement it is called as defect
- Defect report
 - Is a document to maintain all the defects, that test engineer found while test execution
 - The most important deliverables to come out of test. It will have more impact on the quality of the product than most other deliverables from test
 - It is important to write effective defect reports

1.2: Defect Reports

Defect Reporting – The Need

- Emphasize on continuous improvement
- Defect report – an important deliverable
- Inadequate Material
- High impact of defective defect report

1.3 Defect Report
Template

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Project ID:		Project Name :											
2														
3	Defect ID	Module name	Defect Summary	Defect Description	Defect Category	Detected in Browser	Environment	Defect Severity	Defect Priority	Detected in Release #	Detected in Build #	Reported Date	Reported By	Assigned To
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														
16														
17														
18														
19														
20														

Project Profile Defect Tracking Sheet Revision History

Capgemini
ENVELOPING TECHNOLOGY. DISRUPTING.

Copyright © Capgemini 2015. All Rights Reserved 6

- **Defect ID** - A unique number to each defect. This will help to identify the bug record. If you are using any automated bug-reporting tool then this unique number will be generated automatically each time you report the bug.
- **Module Name**- It will contain the name of the module being tested.
- **Defect Summary**- A short summary of the defect. It can be in 1 or 2 lines.
- **Defect Description**- A detailed description of bug. It includes
 1. Clearly mention the steps to reproduce the bug.
 2. How application should behave on above mentioned steps.
 3. What is the actual result on running above steps i.e. the bug behavior.
- **Defect Category**- Categorizing the defect into the appropriate category. The categories considered are:
 - Coding: When the defect is found in the code.
 - Design :When the defect is found in the design
 - Enhancement: If the defect stated is actually an enhancement to the present requirement
 - New Requirement :If the defect stated is actually a new requirement
 - Query : is any question or doubt which might be raised by the tester, it need not be an actual defect.
 - Documentation: is any error found in the documents of the application like the help document etc.
 - Master Data: Is any error found in the master data received
 - Test Review: is while doing the testing any review comments which might be suggested by the tester ,again it may not be a defect itself.

1.3 Defect Report
Template(continued)

	Detected in Release #	Detected in Build #	Reported Date	Reported By	Assigned To	Status	Review type / Test Cycle	Test Case No.	Fixed By	Fixed Date	Verified Date	Verified By	Verified in Release #	Verified in Build #	Attachments	Comments
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																

Project Profile Defect Tracking Sheet Revision History

Capgemini
CONSULTING TECHNOLOGY OUTSOURCING

Copyright © Capgemini 2015. All Rights Reserved 7

- **Detected in Browser** :The name of the browser the defect was found in (IE, Firefox ,etc.)
- **Environment**: Mention the environment where the defect was found. It can be internal Dev,internal QA,External Dev or External QA.
- **Defect Severity** : Impact of the defect on functionality of application. The values it can contain :
 - P0 defect is a defect wherein the main function of the software does not work. e.g. crash, hang, data corruption
 - P1 defect is a defect wherein a function does not work and a tedious work around exists. e.g. One of menu options does not work.
 - P2 defect is a defect wherein the software does useful work but a degree of inconvenience is caused. Correction is not deferrable and easy work around exists.
 - P3 defect is a tolerable defect as corrections are deferrable. E.g. cosmetic problems in user interface like spelling.
 - “High” severity defect is a defect wherein the main function of the software does not work. e.g. crash, hang, data corruption, some of the menu functions do not work
 - “Medium” severity defect is a defect wherein the software does useful work but a degree of inconvenience is caused. Correction is not deferrable and easy work around exists.
 - “Low” severity defect is a tolerable defect as corrections are deferrable. E.g. cosmetic problems in user interface like spelling.

1.4: Defect Reports

Important Attributes

- | | |
|-------------------------|----------------------------|
| ▪ Defect ID | ▪ Assigned To |
| ▪ Module name | ▪ Status |
| ▪ Defect summary | ▪ Review type / Test Cycle |
| ▪ Defect description | ▪ Test Case No. |
| ▪ Defect Category | ▪ Fixed By |
| ▪ Detected in Browser | ▪ Fixed Date |
| ▪ Environment | ▪ Verified Date |
| ▪ Defect Severity | ▪ Verified By |
| ▪ Defect Priority | ▪ Verified in Release # |
| ▪ Detected in Release # | ▪ Verified in Build # |
| ▪ Detected in Build # | ▪ Attachments |
| ▪ Reported Date | ▪ Comments |
| ▪ Reported By | |

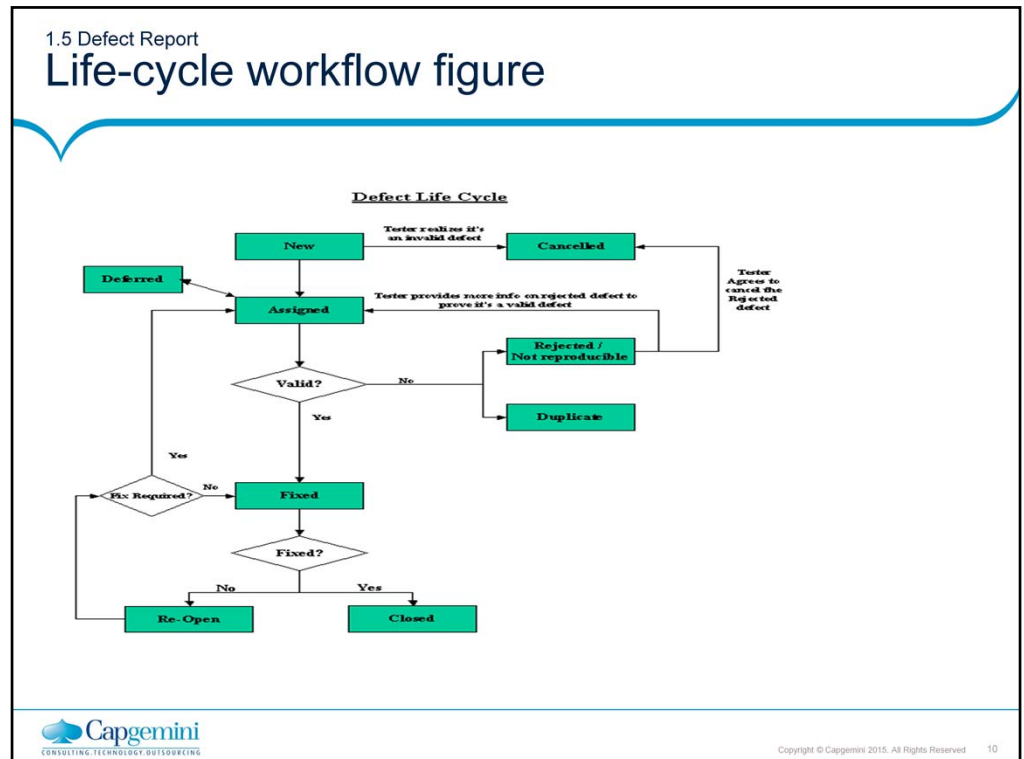


Copyright © Capgemini 2015. All Rights Reserved 8

- **Priority** – It indicates When bug should be fixed? The values are low, medium and high
- **Detected in Release #** : Denotes the Release # in which the defect was detected.
- **Detected in Build #**: Denotes the Build # in which the defect was detected.
- **Reported Date**: The date on which the defect was reported. It need not be same as the date on which it was detected.
- **Reported By**: Name of the person who reported the defect.
- **Assigned To**: Name of the person the defect is assigned to , to fix it.
- **Status**: Gives the status of the defect. It can contain the following values
 - Verified
 - Closed
 - Fixed
 - Active
 - CNR-Could Not be Reproduced
 - NOD: reported bug Not a Defect
 - REP: Repeated bug
- **Review type / Test Cycle**: It can contain following values:
 - PPR-Peer to Peer Review
 - PR-Peer Review
 - SR-Self/Programmer Review
 - QPR-10% Quality Probe(Review)
 - RT-Random Testing
 - UTC-Using Test Cases



- **Test Case No:** Mention the test case No that the defect was found while testing
- **Fixed By:** Name of the person who fixed the defect
- **Fixed Date:** Date on which the defect was fixed
- **Verified Date:** Date on which the defect was verified
- **Verified By:** Name of the person who will be verifying whether the defect has got fixed after the defect is being worked upon
- **Verified in Release #:** Denotes the Release # in which the defect was verified
- **Verified in Build # :** Denotes the Build # in which the defect was verified
- **Attachments:** Attach the proofs showing the defect- screenshots, temp files, etc.
- **Comments:** Should include all the comments made by all touching the defect on the defects till date .It should be captured date wise.



1. **New** – When a Defect is logged and yet to be assigned to a developer. Usually Project Manager or Dev Lead will decide on which defects to be assigned to which developer.
2. **Assigned** – indicates that the developer who would fix the defect has been identified and has started analyzing and working on the defect fix.
3. **Duplicate** – Manager or Developer will update the status of a defect as “Duplicate” if this defect was already reported.
4. **Rejected / Not Reproducible** – This status indicates that the developer is not considering the defect as valid due to following reasons
 - a) Not able to reproduce
 - b) Not a valid defect and it is as per requirement
 - c) Test Data used was invalid
 - d) Defect referring to the Requirement has been de-scoped from the current release, tester was not aware of this late changes.
5. **Deferred** – Defect fix has been held back because of time or budget constraints and project team has got approval from customer to defer the defect till next or future release.
6. **Fixed** – Developer has fixed the defect and has unit tested the fix. The code changes are deployed in test environment for verifying the defect fix.

7. **Reopen** – Status is changed to “Reopen” by a tester, when a tester finds the defect is Not fixed or partially fixed. Developer who fixed the defect looks into the comment that was provided by the tester at the time of reopening the defect. Developer will change the status to “Assigned” and starts working on the fix again. Incase the developer wants the tester to re-verify the defect then he/she will add a comment and will change the defect status to “Fixed”.
8. **Closed** – Tester verifies the defects that are in “Fixed” status and once they find the defect is fixed, they change the status to “Closed”. This is the last status of Defect Life Cycle.
9. **Cancelled** – This status indicates that the tester realized that the defect logged by him was invalid and agreed to cancel it.

a

1.6 Defect Report

Users

- Management
- Maintenance Team lead
- Maintenance Engineer
- Testing Team Lead

1.6 Defect Report

Benefits for Maintenance Team Leads

- Allocate the Defect to the appropriate team member as soon as possible
 - Quickly understand the software version and component responsible for issue
- Effectively prioritize defect for fixing
 - Is it halting the testing process?
 - Are other functionalities dependent on this?
 - Is it important under part release?
 - Expected fixing date
- To accurately take corrective and preventive actions for future developments
 - Category wise, severity wise Defect status (functionality, modules, layers)
 - Average turn around time for Defect fixing
 - Defect density

1.6 Defect Report

Benefits for Maintenance Engineer

- Identify the application version (mainly for products)
 - Isolate the application version when multiple versions are being maintained
- Quickly get to the root cause
 - Concentrate not on symptoms but root cause to isolate the component creating issue
- Know reproducibility and environment/situation of reproducibility
 - Else do not waste time, arrange for necessary dependencies/settings
- Analyze the log file and get clear understanding about the issue
 - Check the exact details - data entered, actions taken, results generated, tables updated (application log, database log)
- Contact the tester who found the Defect
 - To get a first hand information & clarification directly and quickly

1.6 Defect Report

Benefits for Testing Team Lead

- Plan retesting efforts
 - Tentative dates when defects are expected to be fixed
 - Estimated defects
- Analyze quality of Defect reporting process
 - Defect Acceptance Rate, Defect Communication Effectiveness
- Increase accuracy on future estimates
 - Generate accurate summaries for status reporting
 - Application module wise, Severity wise total/open defects
 - Functionality wise, severity wise total/open defect counts
- Monitor performance of the testers
 - Tester wise metrics

1.6 Defect Report

Benefits for the Management

- To know the status of the Defects
 - Application module wise, severity wise defect summary of open and closed defects
 - Category wise severity wise open and closed defects
 - Expected dates for fixing and closing of high severity defects
- To analyze the performance of the teams
 - Team wise – team member wise metrics – count, productivity
- Effective follow-up
 - Generate exception reports – Actions due in next n days, Actions pending for more than n days
 - To take go/no go decision for next cycle/phase/production
 - Defect summary in conjunction with Test Case execution summary
- To know the risks involved
 - Summary and details of known defects (with impact)

1.7 Defective Reports

What is Defective Defect Report?

- Defective defect Report
 - The inaccurate, incomplete and unclear defects results into defective defect report
 - Impact of Defective defect report
 - Wastage of time that is precious in tight schedule
 - Inaccurate / incomplete status leading to wrong / no decisions
 - Inaccurate statistics leading to inaccurate corrective / preventive measures
 - Frustration and ill feeling between development and testing teams

1.8 Defective Reports

The reasons for defective reports

- Cannot reproduce
 - If the maintenance engineer is not able to reproduce the bug, by using steps mentioned in defect report
- Already reported (Duplicate)
- Functionality is as per requirement
- Some one else is responsible
- Additional information needed – Error message detail, Data input, options selected, previous tasks executed etc
- Details provided are not clear
- Some attributes are not provided or not correct – Severity, Transaction Id, version, category etc
- Is a new requirement or change in requirement



Copyright © Capgemini 2015. All Rights Reserved 18

For all above reasons the developer will change the status of the bug as rejected.

1.9 Defective Reports

Root causes for defective reports

- An Assumption - Developer should be able to understand defect quickly and easily with little hint
- Providing all the steps, test data etc. takes lot of time to report
- Testers do not know the importance (usage) of details other than defect description
- Providing evidences for the defect are not considered important
- Informal communication process - through emails, verbal, and historical details are not maintained
- Some defects gets unknowingly fixed due to fixing of other defects
- Features of tool, process not known

1.9 Defective Reports

Other Influencing Factors

- An Requirement of large testing team size
- Large and Complex application architecture with Involvement of multiple development teams
- Teams working from different countries in different time zones
- Business users/testers and developers speaking different languages

1.11 Severe Defects

The Most Severe Defect

- Wrong Severity is a severe defect
 - Importance for go / no go decisions
 - Importance for deferring the defect-fix to next release
 - Credibility of development team is based on such defects

1.11 Defective Reports

Certain Facts

- Quick fixing of some Defects would help first the testers themselves
 - Show stoppers
 - Dependencies and pre-requisites
- Developers are human beings
 - There are bound to be Defects in application
- Varied interests and expectations – High defect count ->
 - Good performance by test team but
 - Bad performance by development team
- The maintenance engineer is not the only user of the Defect report
 - Maintenance team lead, Test team lead, Management

1.10 Defect Free Reports

Recommendations

- Increasing Awareness / Being conscious of
 - Some realities in testing process
 - Users of Defect report
 - Importance of Defect report attributes to different users
- Following process and guidelines
- Preparation before starting the assignment
 - Verify before recording defect
 - Review Defect report before submitting
- Use of tool with required features
- Institutionalizing improvements

1.12 Writing Defect Free Reports

Preparation

- Get complete understanding of Defect Tracking process and tool
 - Guidelines for reporting, checking for duplicate defects, pre-defined definitions for severity, priority, categories etc
- Establish / understand communication protocol
 - Abbreviations and symbols, Providing references to documents, using standards
- Acquire communication skill
 - To provide clear, complete yet concise information about the defect

1.13 Defects

Reporting and Communication Process

- Checking before Recording the defect
 - It is really a Defect
 - It is not yet reported
 - It is a specific or a general issue

1.13 Defects

Reporting and Communication Process (contd.)

- Finding and reporting
 - The root problem
 - The shortest way to recreate the Defect
 - All the other information that can help
 - All evidences – screen shots, references to other documents, correspondences etc
- Ensuring
 - CAN PIG RIDE??



Copyright © Capgemini 2015. All Rights Reserved 26

When in doubt report the defect - Primary objective is to get as less defects (if possible no defects) in production and not high Defect Acceptance Rate

1.13 Defects

Reporting and Communication Process (contd.)

- Condense
- Accurate
- Neutralize
- Precise
- Isolate
- Generalize
- Recreate
- Impact
- Debug
- Evidence



Copyright © Capgemini 2015. All Rights Reserved 27

Key points to make sure the next defect report you write is an effective one.

1. **Condense** - Say it clearly but briefly
2. **Accurate** - Is it a defect or could it be user error, misunderstanding, etc.?
3. **Neutralize** - Just the facts. No zingers. No humor. No emotion.
4. **Precise** - Explicitly, what is the problem?
5. **Isolate** - What has been done to isolate the problem?
6. **Generalize** - What has been done to understand how general the problem is?
7. **Re-create** - What are the essentials in triggering/re-creating this problem?
(environment, steps, conditions)
8. **Impact** - What is the impact to the customer? What is the impact to test?
9. **Debug** - What does development need to make it easier to debug?
(traces, dumps, logs, immediate access, etc.)
10. **Evidence** - What documentation will prove the existence of the error?

It is key to make sure that you have covered the essential items that will be of most benefit to the users of the defect report.

1.13 Reporting and Communication of Defects

Condense - Example

Best Practice	Defect Remark
Don't: Suffers from too much information, most of which is not helpful.	I was setting up the test whose real intent was to detect memory errors. In the process, I noticed a new GUI field that I was not familiar with. I decided to exercise the new field. I tried many boundary and error conditions that worked just fine. I cleared the field and attempted to advance to the next screen, then the program abended. Several retries revealed that anytime, there is not any data for the "product description" field, you cannot advance to the next screen or even exit or cancel without abending.
Do:	The "exit", "Next" or "cancel" functions for the "product information" screen abends when the "product description" field is empty or blank.

1.13 Reporting and Communication of Defects

Neutralize - Example

Best Practice	Defect Remark
Don't: The first clause may be interpreted as a jab at the developer and adds no useful information.	As could have been determined from the original defect, with very little effort, function ABC does indeed abend with any negative value as input.
Do:	Function ABC abends with any negative value. For example: -7, -1, -32767.

1.13 Reporting and Communication of Defects

Precise

Best Practice	Defect Remark
Don't: In this example, it is hard to tell if the problem is (1) the twin-max port not working or (2) printer not returning to ready	Issuing a cancel print when job is in PRT state (job is already in the printer and AS/400 is waiting to receive print complete from printer) cause the twin-max port to not time out. The printer never returns to ready state and indefinitely displays "Printing from Tray1" in the op-panel.
Do: Precede the description with the short summary of exactly what you perceive the problem to be.	Canceling the job, while it is printing causes printer to hang. Issuing a cancel print when job is in PRT state (job is already in the printer and AS/400 is waiting to receive print complete from printer) cause the twin-max port to not time out. The printer never returns to ready state and indefinitely displays "Printing from Tray1" in the op-panel.

1.13 Reporting and Communication of Defects

Generalize

Best Practice	Defect Remark
Don't:	Error message for "file not found" error has garbage characters for the file name
Do:	Error message for "file not found" error has garbage characters for the file name. Every message I tried that expected data to be inserted in the message had the same problem. Messages without inserts were okay.

1.14 Writing Effective Defect Reports

Guidelines

- When you file a defect, it needs to be easily conveyed to the developer
- You must provide clear information
 - Fill out as many fields as you can
 - Provide screen shots, log files, URL's and references to similar defects
 - Include detailed steps to reproduce the issue



Copyright © Capgemini 2015. All Rights Reserved 32

Some more points to remember

- Refer to requirements or design documents where appropriate.
- When possible, check to see if the problem occurred in an earlier build.
- Include data like username, date, time, and details that make the problem easier to find
- If you checked a log file and find nothing, include a comment like "Nothing in [log name].log."
- You may want to try other scenarios and include the results.

Example of good Description:

On the bills page, some amounts have no decimal after the amount, others have two. Since this is currency, we should be consistent and always display two decimals (or whatever is standard) after the amount-- even if the last digit is a zero.

Example of a bad description:

went to page. decimals aren't right

STEPS TO REPRODUCE THE PROBLEM

Provide **easy-to-follow steps** that will reproduce the bug. Include any special setup steps, login information, account information, and any other information that will make it easy to see the problem.

Good example:

Login with username : "user1" and Passwd : "user1"

Click on "Bill details"

Select any order no, It will show you list of items in the order

Click on generate bill.

Bad example:

logged in and went to page. changed field. noticed decimals wrong.

1.15 Using Tools for Reporting

Advantages

- With Primary features
 - Built in validation checks
 - Maintenance of history
 - Generation of summarized information, metrics
 - Ability to quickly search on specified criteria
 - Multiple attachments
 - Reduced other communication issues
 - Online status
- With other optional features
 - Proactive notification when no action taken on due date
 - Involvement of translator for translation when needed
 - Understand number of items on which actions to be taken
 - Change Request approval process

1.16 Project/Organization Level Process

Project/Organization Level Process

- Competency Development of test team in
 - Communication skills
 - Defect management
- Project specific familiarization process
 - Communication protocol
 - Defect reporting and tracking process
 - Defect tracking tool
- Summary / Metrics generation, sharing and monitoring
 - Defect Acceptance Rate
 - Defect Communication Effectiveness
 - $(\text{Total Defects reported} / \text{Number of times Defects are communicated to Maintenance team}) * 100$

1.17 Defect Free Reports

Advantages

- Improved project control
- Improved quality of report
- Improved productivity
- Improved cycle time
- Reduction in overall effort
- On time delivery
- Overall satisfaction
- Improved predictability

1.18 Nine Commitments

Nine Commitments worth making to developers

- We'll test your code as soon as we can after it's built.
- We'll test important things first, and focus on important problems.
- We'll write clear, thoughtful, and respectful problem reports.
- We'll try not to be a bottleneck for development.
- We'll tell you how we're testing, and consider your suggestions.
- We'll look for ways to test better and faster.
- We will not waste your time.
- We will create order out of chaos.
- We will always remember that collectively we win or lose as a team.

Summary

- In this lesson, you have learnt:
 - Defect report is an important deliverable since it gets referred by maintenance team, testing team, management
 - The inaccurate, incomplete and unclear defects results into wrong decisions
 - Follow process and guidelines
 - People fixing defects are most likely to be different than original developers
 - Institutionalizing process and building competencies for defect free defect reporting



Review Question

- Question 1: Which of the following will be entered by test engineer in the defect report
 - Option 1: Reported By
 - Option 2: Resolution Details
 - Option 3: References
 - Option 4: All of the above
- Question 2: Before you log a defect it is not necessary to verify whether it is duplicate because it is time consuming.
 - True/ False
- Question 3: Adding attachments is easy if we are using any tool to log the defect
 - True/ False



Defect Reporting and Defect Life Cycle Management – V 2.0

Lab Book

Document Revision History

Date	Revision No.	Author	Summary of Changes
12/8/09	1	Priya Rane	Revamp
13 /06/11	1.1	Kishori Khadilkar	Revamp
2/6/15	2.0	Alphy Thomson & Shilpa Bhosle	Material Revamp

Table of Contents

<i>Document Revision History</i>	<i>2</i>
<i>Table of Contents</i>	<i>3</i>
<i>Lab 1. Defect Reporting Basics</i>	<i>4</i>
<i>Lab 2. Defect Report.....</i>	<i>5</i>
2:1 Defect Report for Bank Exam Portal Application	5

Lab 1. Defect Reporting Basics

Goals	<ul style="list-style-type: none">• Understand the process of reporting defects.• Learn to report defect free defects.
Time	60 min

1.1 The testing team has found certain defects during testing the Banking Exam Portal Application. As a one of the team member of the testing team the responsibility has been given to you to log these defects into Defect Report so that the development team can understand the defects well and take necessary measures to eliminate the same from the application. You need to ensure that defects found during test execution are logged with proper care so that the defect removal cycle can be reduced.

Note – The below mentioned defects are imaginary defects. Make a proper use of the Defect Tracking Sheet to log below given defects.

Home Page:

1. The link “List of exams on the portal” is not able to navigate the user to the web page displaying various exams available on the Banking Exam Portal application.
2. The link “List of exams on the portal” is displayed at the left side of the web page instead of right side.
3. The User Name field is accepting any type of data.
4. The application is not generating an error even if Password field is left empty while logging to the application
5. The application behaves in a wired manner by disallowing a user to login to the website even if the user has entered the valid credential i.e. valid User Name and Password.
6. After clicking on the Submit button on the home page user gets an error message “Page Cannot Be Displayed”.

Lab 2. Defect Report

Goals	• Understand the application and report defects in defect report
Time	60 min

2:1 Defect Report for Bank Exam Portal Application

Make Defect Entry for the Following Defects in the Defect Entry Report:

Registration:

1. The Spelling of “Personal Information” is wrong. The web page displays it as “Persnal Information”.
2. Mobile number field is accepting alphabets and special characters also.
3. The insert Photo field is accepting a photo of size more than 150 kb.
4. In marital status only 2 options i.e. Married & Unmarried should be shown on the web page, however the web page is displaying three options to the user i.e. Married, Unmarried & Divorcee.
5. After clicking on the “Save & Submit” button, Registration Id should get auto-generated and displayed on the screen; however the application is not able generate and display the Registration ID upon successful registration process.

2.2 Defect Report for Bank Exam Portal Application – “Add Exam Details”

You have designed test cases for testing the various functionalities of “**Add Exam Details**” page. Assume that your test cases have failed. Prepare the defect report for all failed test cases.

Note: Every participant may have different test cases. Use the test cases that you have written for logging the defect in the defect report.

Do not restrict the scenarios to a single page or field; rather focus on the flow of the application as an end user or applicant.