Name:- Shah miloni A.
Rollno:- 92
Section:- T4-B

Date / /
Page 1

Unit-1

* Definition of Software Testing :-

- Testing is the process of executing a program with the intent of finding errors

- Testing can show the presence of bugs but never their absence.

- Testing is a support function that helps developers look good by finding their mistakes before anyone else does

- Software testing is a process that detects important bugs with the objective of having better quality software

* Terms :-

⟹ Failure, fault, defect, Bug/error :

→ Failure :

- Failure means the in ability of our system or component to perform a required function according to its specification.

Scanned by CamScanner

- when result or behavious of the system under test are different as compared to specified expectations then failure exists.

→ **Fault / Defect / Bug :**

- Failure is the term which is used to describe the problems in a system on the output side.

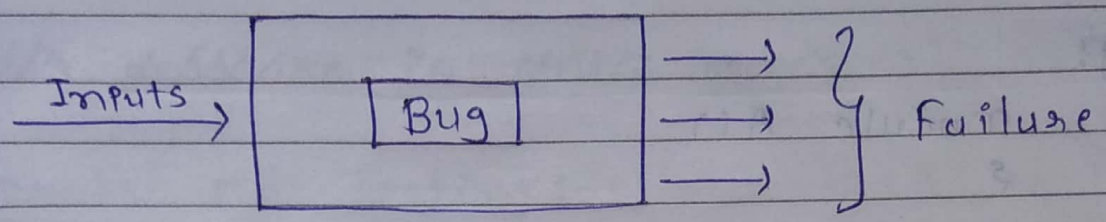
- fault is a condition that actually causes our system to produce failure.

- fault is also called defect or Bug.

- one failure maybe due to one or more bugs and one bug may cause one or more failure. Thus when a bug is executed then failures are generated.

- Some bugs are hidden in the sense this are not executed, as they don't get the required conditions in the system. So hidden bugs may not always produce failures. They may execute only in certain rare condition.

Scanned by CamScanner



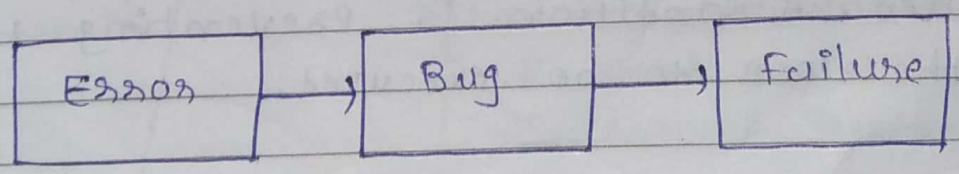
Testing failure.

→ **Error:**

- whenever a development ~~in~~ team member makes a mistake in any phase of SDLC, errors are produce.

- It might be a typographical error, misleading of a specification, a misunderstanding of what a subroutine does, and so on.

- Error is a very general term used for human mistakes. Thus an error causes a bug and the bug in ~~term~~ turn causes failures.



flow of faults

Scanned by CamScanner

- eg:

```
module A()
{
    ...
    while (a > n+1);
    {
        ...
        Print ("value of x is", x);
    }
    ....
}
```

- Suppose, the module shown about is expected to print the value of x but when this module is executed, the value of x is not printed. It is a failure of the program.

- when we try to find the reason of failure, we find that the while loop is not the executed. A condition is preventing the body of the loop to be executed.

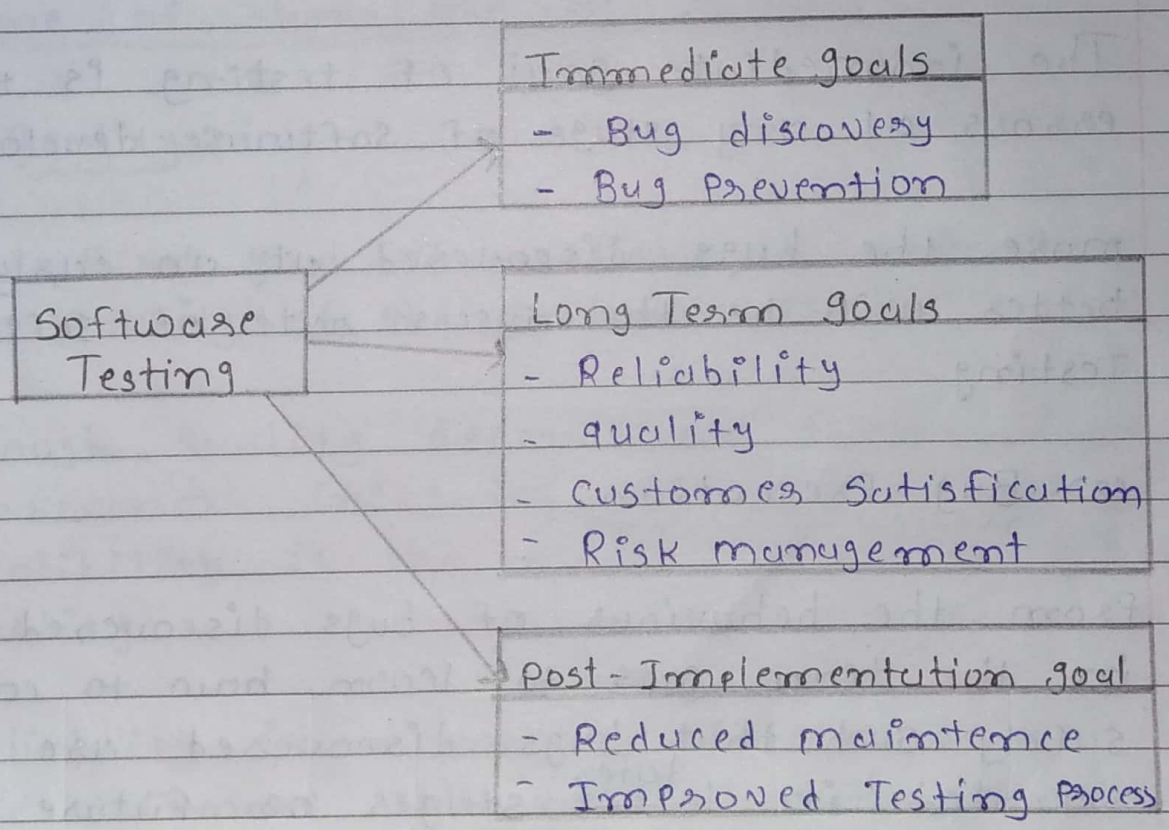
- This is known as Bug / defect / fault. on close observation, we find a semicolon being displayed after the while loop and it not allowing the loop to execute. This mistake is called error.

Scanned by CamScanner

- This mistake is called error.

* Goals of Testing :-

- The goals of Software Testing may be classified into three major categories as shown in the following figure.



Scanned by CamScanner

⟹ Short term or Immediate goals:

- These goals are the immediate result after performing testing.

→ Some of them are discussed below:

(1) Bug discovery:

- The immediate goal of testing is to find errors at any stage of software development.

- more the bugs discovered at an early stage, better will be the success rate of software Testing.

(2) Bug Prevention:

- From the behaviour of bugs discovered, everyone in the team gets to learn how to code safely such that bugs discovered are not repeated in ~~letter~~ later stages or future projects.

- Though errors can not be eliminated totally but they can be minimized.

Scanned by CamScanner

- In this sense bug prevention is a superior goal of testing.

⟹ Long term goals:

- These goals affect the product quality in the long run

→ Some of them are as follows:

(1) Quality:

- The first goal of testing is to improve the quality of the software product.

- Though quality depends on factors like correctness, integrity, efficiency etc, reliability is the major factor to achieve quality.

- reliablity is a matter of confidence that the software will not fail and these level of confidence increases with rigourous testing.

Scanned by CamScanner

- These confidence is reliability, interm increases the quality.

(2) Customer Satisfication:

- If we want the customer to be satisfied with the software Product then testing should be complete & through

- Testing must satisfied the user on all the specified requirements mention in the user manual, as well as for the unspecified requirements which are otherwise understood.

- A complete testing Process achieve reliability enhances the quality and quality increases customer Satisfuction.

(3) Risk management:

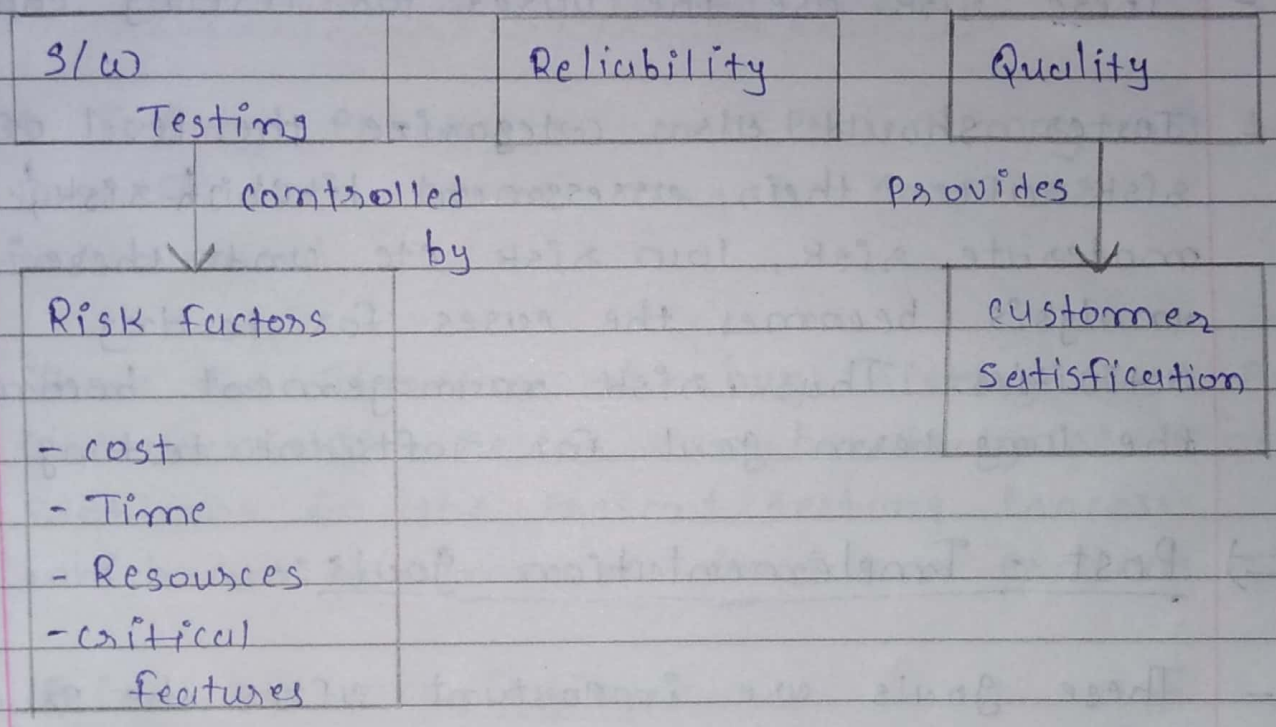
- Risk is the probability that undesirable events will occur in a system.

- These undesirable events will prevent the organsation from successfully implementing its business initiatives.

Scanned by CamScanner

- Risk must be control to manage them with ease.

- Software testing may act as a control which can help in eliminating or minimizing risk.



- The purpose of software testing as a control is to provide information to management so that they can better react to risk situations

- Testing may indicate that the software being develop can not be delivered on time or high priority bugs will not be by the specified time. with these advance information decision can be made to

Scanned by CamScanner

minimize risk situation.

- It is testers responsibility to evaluate business risk such as cost, time, resources and critical features.

- These risk are the bases for testing choices.

- Tester should also categorize the level of risk after their assessment likehigh-risk, moderate risk, low risk etc and these analysis becomes the cases for testing activities. Thus, risk management becomes the long term goal for software testing.

⟹ Post - Implementation goals :-

- These goals are important after the product is released.

→ Some of them are,

(1) Reduce maintenance cost :

- The only maintenance cost in a software product is it's failure due to errors. ~~Post~~

Scanned by CamScanner

- Cost release error are costlier to fix as they are difficult to detect. Thus, if testing is done rigorously then chances of failure are minimized which can reduce the maintenance cost.

e2) Improved S/w Testing Process:

- A Testing Process or one Project may not be successful and there may be scope or improvement

- ~~There for~~ Therefore the bug history and post implementation result can be analysied findout problems in the present testing process which can be improve in future projects

* Principles of Testing :-

1. effictive testing, not exhaustive testing:

- All possible combination of test become so large that it is impractical to test them all. so considering the domain of testing as infinite, exhaustive testing is not possible.

Scanned by CamScanner

- Therefore the tester approach should be based on effective testing to properly cover program logic and all conditions in the component level design.

2. Testing is not a single phase performed in SDLC :

- The testing phase after coding in SDLC is apart of the whole testing process.

- Testing process starts as soon as the specifications of the systems are prepared and it continous till the release of the product.

3 Destructive approach for constructive testing :

- The criteria to have a successful testing is to discover more and more bugs, and not to show that the system does not contain any bugs.

4. Early testing is the best policy :

- Testing starts as soon as requirements specification are prepared.

Scanned by CamScanner

- The cost of bugs can be reduced tenfold (ten times) in early testing as bugs are harder to detect in later stages if they go undetected.

- Thus the policy in testing is to start as early as possible.

5 Possibility of existence of an error in a section of program is proportional to the number of errors already found in that section:

- If some sections are found error-prone in testing then our next testing effort should give priority to these error-prone section.

6. Testing strategy should start at the smallest module level and expand towards the whole program:

- Testing can not be perform directly on the whole system.

- It must start with the individual modules and slowly integrate the modules and test them after these the whole system should be tested.

Scanned by CamScanner

7. Testing should also be perform by an independent team:

- Programmers are biased (पक्षपाती) towards the correct functionning of the specified requirements and not towards detecting bugs.

- Therefore, it is always recommended you have the software tested by an independent testing team.

8. Everything must be recorded in software Testing:

- Testing is a planned process.

- It demands that every detail be recorded and documented.

- It must have the record of every test case run and the bugs reported.

- executing the test-cases in a recorded and documented way can greatly help while observing the bugs.

- moreover, observation can be a lesson for other projects.

Scanned by CamScanner

9. Invalid inputs and unexpected behaviours have a high probability of finding an error:

- whenever the software is tested, we test for valid inputs and for the functionality that the software it supposed to do. but thinking in a negative way, we must test the software with invalid inputs and the behaviour which is not expected in general.

- This is also a part of effective testing.

10. Testers must participate in specification and design reviews:

- Testers role is not only to get the software and documents or testing.

- If they are not participating in other reviews like specification and design, it may be possible that either some specifications are not tested or some test cases are built for no specifications.

Scanned by CamScanner

* Role of Testing:-

- Testing plays a vital role in software development.

- Testing is an important and valuable stage in the SDLC.

- Software Testing is not an easy task. each & everyday there will be challenges in coding as well as decoding.

- Software Testing has now become a part of Programming so that the developers are able to rectify errors right from the beginning stage.

- It is a stage where the developers find bugs in the software and make the software bug free.

- when Testing is introduce in the early stage, the cost of fixing the bug can be reduced.

- cost of fixing the bug becomes larger when a bug is not found at the right time.

Scanned by CamScanner

→ we role of software testing is significant in following areas:

(1) improving the software:

- The role of testing in software developement begins with in proved realibility, quality and performance of the software.

- It assists a developer to check out whether the software is performing the right way and to assure that the software is not performing what it is not supposed to do.

(2) Quality assurance:

- The software tester plays a key role in SDLC for producing a quality products.

- Testing will be neccessary for making the software defect free.

- defects can be identify and rectified.

- Testing froms an integral part of the SDLC for improving consistency and performance of software.

Scanned by CamScanner

- Software testing services help we developer to know the difference between actual and expected. ~~out come~~ outcome which helps to improve the quality of the product

(3) Helps to avoid dengerous situation:

- without proper testing, produced software can be dengerous to the user.

- They will be confused and lose credibility towards that particular software

- A good and effective software testing helps to improve security. Authentication and validation are the main aims of software testing services.

- Basically software testing helps not only to discover defects in software but also its configuration

Scanned by CamScanner

Imp

* Software Testing Life cycle (STLC):-

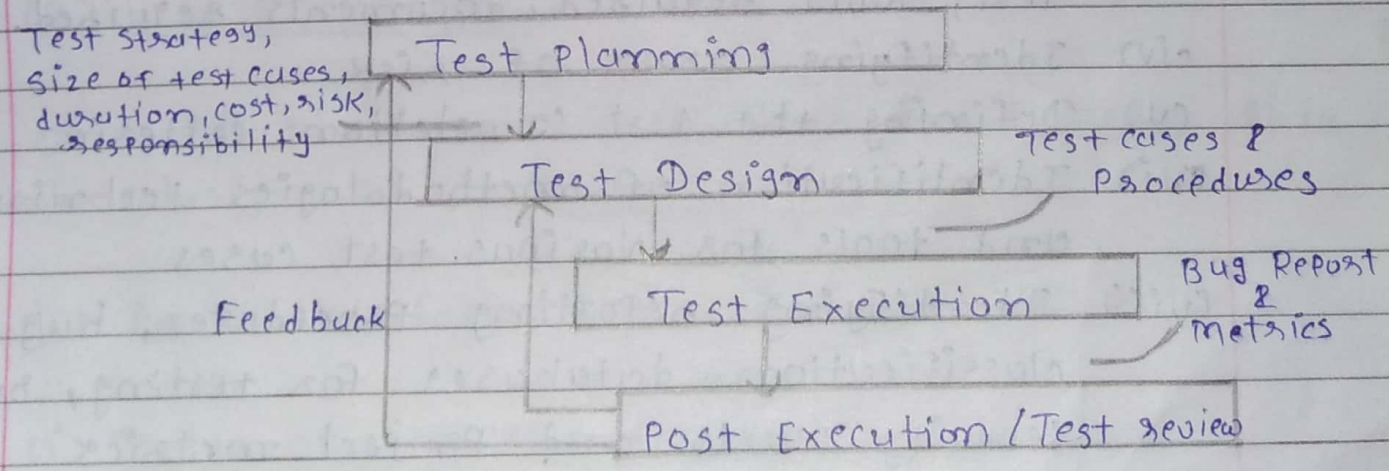
- The testing process divided into a well defined sequence of steps. is termed as Software testing life cycle (STLC).

- The major contribution of STLC is to involve the testers at early stages of development.

- This as a significant benefit in the project schedule and cost

- The STLC also helps the management in measuring specific milestones.

→ STLC consists of the following phases:



STLC

Scanned by CamScanner

(1) Test planning :-

- The goal of test planning is to take into account the important issues of testing strategy namely resources, schedules, responsibilities risk and priorities as a roadmap

- Test planning issues are in tune with the overall project planning.

→ Broadly following are the activities during test planning:

(i) Defining the test strategy
(ii) estimate the number of test cases, their duration and cost
(iii) Plan the resources like the manpower to test, tools required, documents required.
(iv) Identifying areas of risk
(v) Defining the test completion criteria
(vi) Identification of methodologies techniques and tools for various test cases
(vii) Identifying reporting procedures, bug classification, databases for testing, bug severity levels and project metrics

Scanned by CamScanner

- The major output of test planning is the test plan document.

→ Test plans are develop for each level of testing. after Analysing the issues, the following activities are performed:

(i) Develop a test case format
(ii) Develop test case plans according to every phase of SDLC
(iii) Identify test cases to be automated
(iv) Prioritize the test cases according to their importance and criticality
(v) Define areas of stress and performance testing.
(vi) Plan the test cycles required for regression testing

(2) Test Design :-

- one of the major activities in testing. this the design of testcases.

→ It improved following critical activities.

(i) Determining the test objectives and these prioritization:

Scanned by CamScanner

- The test objectives reflect the fundumental elements that need to be tested to ~~sta~~ satisfied an objective.

- For this purpose, unit together reference materials like software requirements specification and design documentation.

- Then on the bases of the reference materials, a team of experts compile a list of test objectives

- This list should also be prioritized depending upon the scope the risk.

(ii) Preparing list of items to be tested:

- The objectives thus obtained are now converted into list of items, that are to be tested under an objective.

(iii) mapping items to test cases:

- After making a list of items to be tested, there is a need to identify the test cases

Scanned by CamScanner

- A matrix can be created for this purpose, identifine which test case will be covered buy which items.

- The existing test cases can also be used for this mapping.

- This matrix will help in:

(a) identify the major test scenarios.
(b) identifine and reducing the redundant test cases
(c) identifine the absence of a test case for a particular objective and as a result, creating them.

- This is only one rule in designing testcases. covered all features, but do not make too many testcases.

- Some Attributes of a good testcase are given below:

(a) The testcase should be design keeping in view the criticality and high risk requirement in order to place a greater priority upon.

Scanned by CamScanner

(b) The testcase should be design such that there is a high probability of finding an error.

(c) each test case should address a unique functionality there by not wasting time and resources

(d) A testcase should be modular so, that there is no complexity and it can be re-used to execute various functional paths.

(iv) Selection of testcase design techniques:

- while designing test cases, there are too broad categories namely blackbox testing and whitebox testing

- blackbox testing generate test cases without knowing the internal working of a system. where-as whitebox testing is used to find errors in the internal logic of the system.

- The techniques to design testcases are selected such that there is more coverage and the system detects more bugs.

Scanned by CamScanner

(v) Creating test cases and test data:

- The test cases mention the objectives under which a test case is being design, the inputs required and the accepted outputs

- While giving input specifications test data must also be choosen with care, as this may lead to incorrect execution of test cases.

(vi) Setting of the test environment & supporting tools:

- The test created about needs some environment settings and tools. so, details like hardware configurations, testers, interfaces, operating systems and manuals must be specifine during this phase.

(vii) Creating test procedure specification:

- This is a description of how the testcase will be run.

- It is in the form of sequence steps.

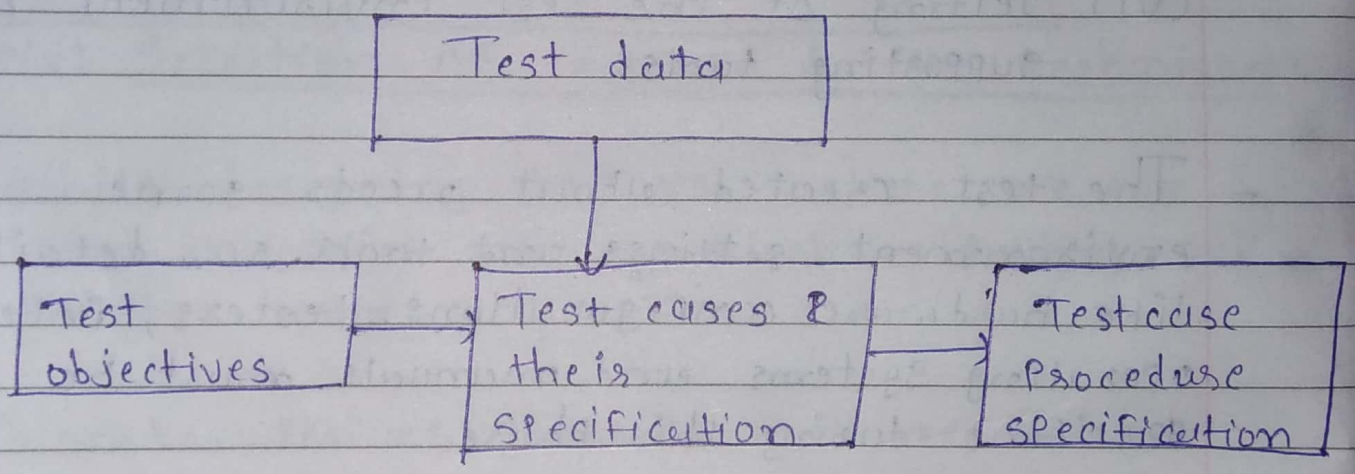
- This procedure is used by the testers at the time of execution of test cases.

Scanned by CamScanner

⟹ Conclusion :

- All the details specified in the test design phase are documented in the test designs specification.

- This document provides the details of the import specifications, output specification, environmental needs and other procedural requirement for the test cases.



Test case Design Steps.

(3) Test Execution :-

- In this phase, all test cases are executed including verification & validation.

Scanned by CamScanner

- Verification test cases are started at the end of each SDLC Phase.

- Validation test cases are started after the completion of a module.

- It is the decision of the test key too opt four automation or manual execution.

- Test results are documented in the test incident reports, test logs, testing status and test summary reports.

- Responsibilities at various levels for execution of testcases are outlined in the following table.

Test execution level	Responsibility
- Unit	developer of the module.
- Integration	Testers & developers
- System	Testers, developers, Endusers
- Acceptence	Testers, endusers

Scanned by CamScanner

(4) Post - Execution / Test Review :-

- After successful test execution, bugs will be reported to the concerned developer.

- These phase is to analys bug related issues and get feedback so that maximum number of bugs can be removed.

→ As soon as the developer gets the bug report, he performs following activities :

(i) Understanding the bug:

- The developer analyses the bug reported and builds a understanding of its whereabouts.

(ii) Reproducing the bug:

- Now, he conforms the bug by reproducing the bug and the failure that exists.

- This is necessary to cross check failures

(iii) Analysing the nature ~~of~~ and cause of the bug:

- After examining the failures of the bug, the developer starts debugging its symptoms and

Scanned by CamScanner

tracks back to the actual location of the error in the design.

- After fixing the bug, the developer reports to the testing team and the modify portion of the software is tested once again.

- After these, the results from manual and automated testing can be collected.

- The final bug report and associated matrics are reviewed and analysed for overall testing process.

→ The following activities can be done

(a) ~~Follow~~ Reliability analyses:

- It can be perform to establish whether the software meets the pre-defined reliability goals or not.

- If so, the product can be released. if not, then the time & resources required to reach the reliability goals are outlined

Scanned by CamScanner

(b) Coverage analyses:

- It can be used as an alternative criteria to stop testing

(c) Overall defect analyses:

- It can identify risk areas and help focus our efforts on quality improvement.

Imp
* Verification & Validation :-

1. What is verification?
- The verification process includes checking documents, design, code and program.

2. What is validation?
- Validation is a dynamic mechanism of software testing to validate the actual product

3. ~~DRAA~~

Scanned by CamScanner

3. Difference between verification & validation

Verification	Validation
(i) The verification process includes checking documents, design, code & program.	validation is a dynamic mechanism of software testing to validate the actual product
(ii) It does not involve executing the code.	It always involves executing the code.
(iii) It is uses methods like reviews, walk throughs, desk-checking, inspections etc.	It uses methods like blackbox testing, whitebox testing & ~~know~~ non-functional testing.
(iv) It finds bugs early in the development cycle	It can find bugs that the verification process cannot catch.
(v) Target is application & software architecture, specification, High level design, database design etc.	Target is the actual product

Scanned by CamScanner

(vi)	QA team does verification & make sure that the software is as per the requirements in the SRS document	with the involvement of testing team, validation is executed on software code
(vii)	It comes before validation	It comes after verification

4. example of verification & validation:

- In software engineering, consider the following specification.
  "A clickable button with name submet"

- verification would check the design document & correct the spelling mistake.

- Otherwise the development team will create a button like

  [Submet]

- So new specification is,
  "A clickable button with name Submit"

- once the code is ready, validation is done.

Scanned by CamScanner

- Validation test found button not clickable.
  [Submit]

- due to validation testing, the developement team will make the submit button clickable.

* V-testing Life cycle (SDLC-V model):-

- Refer notes on classroom

* Static & Dynamic testing:-

⇒ What is static testing:

- Under static testing, code is not executed. rather it manually checks the code, requirement documents and design documents to find errors. hence, the name "static".

- The main objective of this testing is to improve the quality of software products by finding errors in the early stages of the development cycle.

- This testing is also called a non-execution technique or varification testing.

Scanned by CamScanner

- Static testing involves manual or automated reviews of the documents.

- This review is ~~sum~~ done during ~~and~~ an initial phase of testing to catch defect early in the STLC.

- It examins work documents & provides review comments.

→ Examples of Work documents

(i) Requirement specifications
(ii) Design document
(iii) Source code
(iv) Test Plans
(v) Test cases
(vi) Test scripts
(vii) help or user document
(viii) web page contain

⟹ <u>What is Dynamic testing</u> :

- Under dynamic testing, a code is executed. It checks for functional behaviour of software system, memory / CPU usage and overall Performance of the system. hence the name dynamic.

Scanned by CamScanner

- The main objective of these testing is to confirm that the software product works as per the business requirements.

- These testing is also called an execution technique or validation testing

- Dynamic testing executes the software and validates the output with the expected result.

- Dynamic testing is perform at all levels of testing and it can be either black or white box testing

⟹ Difference between Static & dynamic testing:

Static Testing	Dynamic Testing.
(i) Testing is done without executing the program	Testing is done by executing the program
(ii) This testing does the varification process	This testing does the validation process
(iii) Static testing is about prevention of defects	Dynamic testing ~~gives~~ is about finding and fixing the defects

Scanned by CamScanner

(iv)	Static testing gives an assessment of code and documentation	Dynamic testing gives bugs in the software system
(v)	Static testing involves a checklist and process to be follow	Dynamic testing involves testcases for execution
(vi)	This testing can be perform before compilation	Dynamic testing is perform after compilation
(vii)	static testing covers the structural & statement coverage testing	Dynamic testing covers the executable file of the code
(viii)	cost of finding & fixing defects is less	cost of finding & fixing defects is high
(ix)	Return on investment will be high as these process takes place at an early stage.	Return on investment will be low as these process takes place after the development phase
(x)	more reviews are highly recommended for good quality	more defects are highly recommended for good quality

Scanned by CamScanner

(xi) Requires loads of meetings	Comparatively requires lesser mettings

⇒ Conclusion :

- Verification & Validation are to measures used to check that the software product ~~meets~~ needs the requirements specifications

- Static testing involves verification whereas dynamic testing involves validation.

- Together they help in improving software quality.

Scanned by CamScanner