Section-C

Lecture-21

Dronacharya College of Engineering

Introduction to Software Testing

What is Software Testing

Several definitions:

"Testing is the process of establishing confidence that a program or system does what it is supposed to." by Hetzel 1973

"Testing is the process of executing a program or system with the intent of finding errors." by Myers 1979

"Testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results."
by Hetzel 1983

Software Testing

Software testing is a process used to identify the correctness, completeness and quality of developed computer software.

It is the process of executing a program / application under positive and negative conditions by manual or automated means. It checks for the :-

- Specification
- Functionality
- Performance

Testing Objectives

The Major Objectives of Software Testing:

- Uncover as many as errors (or bugs) as possible in a given timeline.

- Demonstrate a given software product matching its requirement specifications.

- Validate the quality of a software testing using the minimum cost and efforts.

- Generate high quality test cases, perform effective tests, and issue correct and helpful problem reports.

Major goals:

uncover the errors (defects) in the software, including errors in:

- **requirements from requirement analysis**
- **design documented in design specifications**
- **coding (implementation)**
- **system resources and system environment**
- **hardware problems and their interfaces to software**

What ….????

…is an ”ERROR”??

….is a ”Bug”??

….is Fault, Failure ??

Bug, Fault & Failure

A person makes an **Error**
That creates a **fault** in software
That can cause a **failure** in operation

Error : An error is a human action that produces the incorrect result that results in a fault.

Bug : The presence of error at the time of execution of the software.

Fault : State of software caused by an error.

Failure : Deviation of the software from its expected result. It is an event.

Who does Software Testing

- Test manager
    - manage and control a software test project
    - supervise test engineers
    - define and specify a test plan

- Software Test Engineers and Testers
    - define test cases, write test specifications, run tests

- Independent Test Group

- Development Engineers
    - Only perform unit tests and integration tests

- Quality Assurance Group and Engineers
    - Perform system testing
    - Define software testing standards and quality control process

Who is a Software Tester??..

Software Tester is the one who performs testing and find bugs, if they exist in the tested application.

Software Testing Activities

- Test Planing

Define a software test plan by specifying:

- a test schedule for a test process and its activities, as well as assignments
- test requirements and items
- test strategy and supporting tools

- Test Design and Specification

- Conduct software design based well-defined test generation methods.
- Specify test cases to achieve a targeted test coverage.

- Test Set up:

- Testing Tools and Environment Set-up

- Test Operation and Execution

- Run test cases manually or automatically

Software Testing Activities

- Test Result Analysis and Reporting

Report software testing results and conduct test result analysis

- Problem Reporting

Report program errors using a systematic solution.

- Test Management and Measurement

Manage software testing activities, control testing schedule, measure testing complexity and cost

- Test Automation

- Define and develop software test tools
- Adopt and use software test tools
- Write software test scripts and facility

Software Testability

Software testability means how easily a computer program can be tested. There are certain metrics that can be used to measure testability
Following are some key characteristics of testability.

1. **Operability:** ***the better it works, the more efficient is testing process***

2. **Observability:** ***what you see is what you test***

3. **Controllability:** ***the better it is controlled , the more we can automate the testing process***

4. **Decomposability:** ***by controlling the scope of testing, we can more quickly isolate problems and perform smarter testing.***

5. **Simplicity:** ***the less there is to test, the more quickly we can test it.***

6. **Stability:** ***the fewer the changes .***

7. **Understandability:** ***the more information we have ,the smarter we will***

Software Testing Principles

- Principle #1: Complete testing is impossible.

- Principle #2: Software testing is not simple.
  - Reasons:
    - Quality testing requires testers to understand a system/product completely
    - Quality testing needs adequate test set, and efficient testing methods
    - A very tight schedule and lack of test tools.

- Principle #3: Testing is risk-based.

- Principle #4: Testing must be planned.

- Principle #5: Testing requires independence.

- Principle #6: Quality software testing depends on:
    - Good understanding of software products and related domain application
    - Cost-effective testing methodology, coverage, test methods, and tools.
    - Good engineers with creativity, and solid software testing experience

Software Testing Principles

- Principle #7: All test should be based on customer requirements
- Principle #8: Software testing should be planned long before testing begins
- Principle #9: Document test cases and test results.
- Principle #10: Use effective resources to test.

Software Testing Myths

- We can test a program completely. In other words, we test a program exhaustively.

- We can find all program errors as long as test engineers do a good job.

- We can test a program by trying all possible inputs and states of a program.

- A good test suite must include a great number of test cases.

- Good test cases always are complicated ones.

- Software test automation can replace test engineers to perform good software testing.

- Software testing is simple and easy. Anyone can do it. No training is needed.

Importance Of Testing In SDLC

When to Start Testing in SDLC

- Requirement
- Analysis
- Design
- Coding
- Testing
- Implementation
- Maintenance

- ***Testing starts from Requirement Phase***